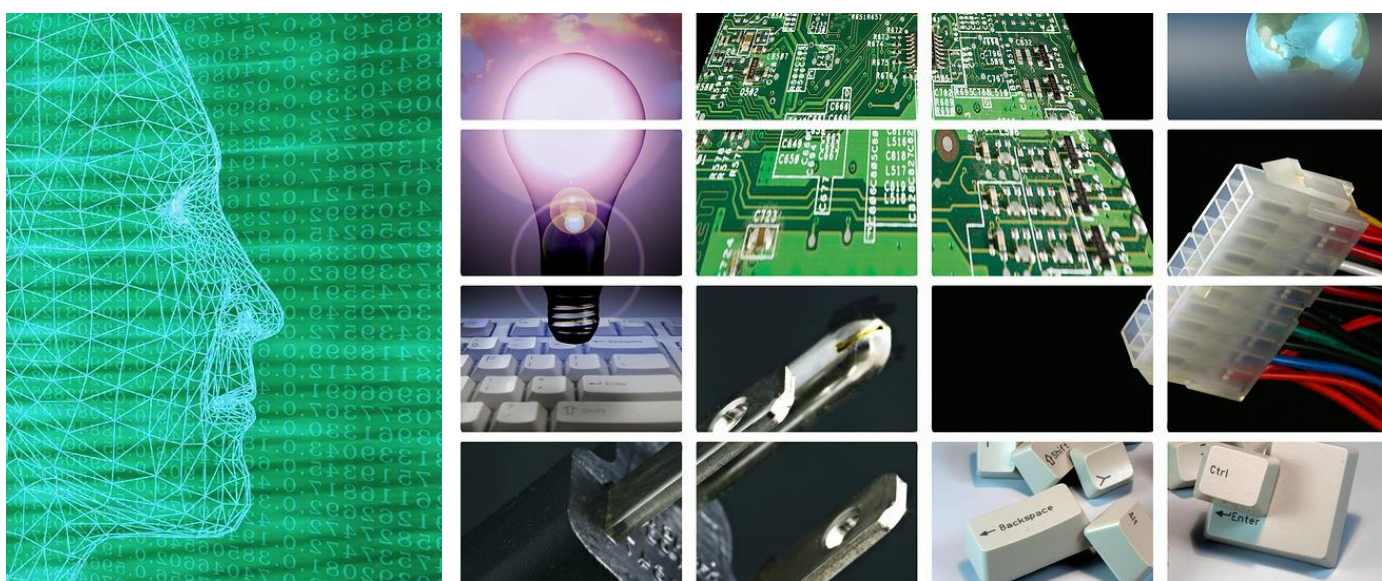


# КОМПЮТЪРНИ НАУКИ И ТЕХНОЛОГИИ



FCA

Bulgaria Communication Chapter

Faculty of Computing & Automation

# COMPUTER SCIENCE AND TECHNOLOGIES

# Компютърни науки и технологии

## Издание

на Факултета по изчислителна техника и  
автоматизация  
Технически университет - Варна

**Гл. редактор:** доц. д-р Ю. Петкова

## Редакционна колегия:

проф. д.н. Л. Личев (Острава)  
проф. д.н. М. Илиев (Русе)  
проф. д-р М. Лазарова (София)  
проф. д-р Г. Спасов (Пловдив)  
проф. д-р Т. Ганчев (Варна)  
доц. д-р П. Антонов (Варна)  
доц. д-р Р. Вробел (Вроцлав)  
доц. д-р Н. Атанасов (Варна)  
доц. д-р М. Маринов, (Русе)

**Печат:** ТУ-Варна

## За контакти:

Технически университет - Варна  
ФИТА  
ул. „Студентска” 1, 9010 Варна,  
България  
тел./факс: (052) 383 320  
E-mail: [yulka.petkova@tu-varna.bg](mailto:yulka.petkova@tu-varna.bg)

**ISSN 1312-3335**

# Computer Science and Technologies

## Publication

of Computing and Automation Faculty  
Technical University of Varna

**Chief Editor:** Assoc. Prof. Y. Petkova, PhD

## Editorial Board:

Prof. L. Lichev, DSc (Ostrava)  
Prof. M. Iliev, DSc (Ruse)  
Prof. M. Lazarova, PhD (Sofia)  
Prof. G. Spasov, PhD (Plovdiv)  
Prof. T. Ganchev, PhD (Varna)  
Assoc. Prof. P. Antonov, PhD (Varna)  
Assoc. Prof. R. Wrobel (Wroclaw)  
Assoc. Prof. N. Atanasov, PhD (Varna)  
Assoc. Prof. M. Marinov, PhD (Ruse)

**Printing:** TU-Varna

## For contacts:

Technical University of Varna  
Faculty of Computing and Automation  
1, Studentska Str., 9010 Varna,  
Bulgaria  
Tel/Fax: (+359) 52 383 320  
E-mail: [yulka.petkova@tu-varna.bg](mailto:yulka.petkova@tu-varna.bg)

**ISSN 1312-3335**

## СЪДЪРЖАНИЕ

## CONTENTS

1	<b>МОДЕЛ ЗА ПРОЕКТНО-БАЗИРАНО ОБУЧЕНИЕ В КОНТЕКСТА НА УМНИТЕ ГРАДОВЕ</b> <i>Николай Кр. Николов</i>	5	<b>A MODEL FOR PROJECT-BASED LEARNING IN THE CONTEXT OF SMART CITIES</b> <i>Nikolay Kr. Nikolov</i>	1
2	<b>РАЗРАБОТВАНЕ НА ИНТЕЛИГЕНТНА СИСТЕМА ЗА УПРАВЛЕНИЕ НА ПОТРЕБЛЕНИЕТО НА ЕНЕРГИЯ В ДОМА</b> <i>Николай Кр. Николов</i>	11	<b>DEVELOPMENT OF AN INTELLIGENT SYSTEM FOR MANAGING ENERGY CONSUMPTION IN THE HOME</b> <i>Nikolay Kr. Nikolov</i>	2
3	<b>ИЗСЛЕДВАНЕ НА НАЙ-ВИСОКАТА ВЪЗМОЖНА СКОРОСТ ПРИ БАЗИРАНОТО НА FPGA ОТКРИВАНЕ НА КОНТУРИ С ИЗПОЛЗВАНЕ НА ГРАДИЕНТИ: ТЕХНОЛОГИЯ НА ДЕФИНИРНЕ НА КОНТУРИТЕ С ХИСТЕРЕЗИС БЕЗ РЕКУРСИВНИ ИЗЧИСЛЕНИЯ</b> <i>Димитър Кромичев</i>	21	<b>INVESTIGATING THE ULTIMATE EXECUTION SPEED IN FPGA BASED GRADIENT EDGE DETECTION: TECHNOLOGY OF NON-RECURSIVE HYSTERESIS THRESHOLDING</b> <i>Dimitre Kromichev</i>	3
4	<b>ТЕХНОЛОГИЯ ЗА ОПТИМИЗИРАНЕ НА СКОРОСТТА ПРИ БАЗИРАНОТО НА FPGA ОТКРИВАНЕ НА КОНТУРИ С ИЗПОЛЗВАНЕ НА ГРАДИЕНТИ: ИЗТЪНЯВАНЕ НА КОНТУРИТЕ С ПАРАЛЕЛНО ИЗЧИСЛЯВАНЕ НА ПРАГОВАТА СТОЙНОСТ</b> <i>Димитър Кромичев</i>	31	<b>TECHNOLOGY OF OPTIMIZING THE SPEED IN FPGA BASED GRADIENT EDGE DETECTION: PARALLEL EDGE THINNING AND THRESHOLD COMPUTATIONS</b> <i>Dimitre Kromichev</i>	4
5	<b>ПОСТИГАНЕ НА НАЙ-ВИСОКА ВЪЗМОЖНА СКОРОСТ ПРИ БАЗИРАНОТО НА FPGA ОТКРИВАНЕ НА КОНТУРИ С ИЗПОЛЗВАНЕ НА ГРАДИЕНТИ: ТЕХНОЛОГИЯ ЗА ТОЧНО ИЗЧИСЛЯВАНЕ НА ПОСОКАТА НА ГРАДИЕНТА</b> <i>Димитър Кромичев</i>	41	<b>ULTIMATE EXECUTION SPEED IN FPGA BASED GRADIENT EDGE DETECTION: TECHNOLOGY OF ACCURATE COMPUTATION OF GRADIENT DIRECTION</b> <i>Dimitre Kromichev</i>	5
6	<b>МОДЕЛИРАНЕ НА ЙЕРАРХИЧНА СТРУКТУРА ЗА НОРМАТИВНА УРЕДБА ПО ИНФОРМАЦИОННО И КОМУНИКАЦИОННО ОСИГУРЯВАНЕ</b> <i>Георги Николов</i>	51	<b>MODELLING OF A HIERARCHICAL STRUCTURE FOR LEGAL REGULATION ON INFORMATION AND COMMUNICATION PROVISION</b> <i>Georgi Nikolov</i>	6

7	<b>АВТОМАТИЗИРАН ПРОЦЕС ЗА РЕЗЕРВИРАНЕ ПРИ ИНТЕГРИРАН ОБМЕН НА СЕЛЕКТИРАНА ИНФОРМАЦИЯ</b> <i>Георги Николов</i>	66	<b>AUTOMATED RESERVATION PROCESS WITH INTEGRATED EXCHANGE OF SELECTED INFORMATION</b> <i>Georgi Nikolov</i>	7
8	<b>МЕТОДИКА ЗА ОБУЧЕНИЕ НА АСЕМБЛЕРЕН ЕЗИК</b> <i>Жейно Жейнов, Десислава Ангелова</i>	80	<b>ASSEMBLY LANGUAGE TEACHING METHODOLOGY</b> <i>Zhejno Zhejnov, Desislava Angelova</i>	8
9	<b>ПРИЛОЖЕНИЕ НА СИМЕТРИЧЕН КРИПТОГРАФСКИ АЛГОРИТЪМ AES НА 8 БИТОВ PIC МИКРОКОНТРОЛЕР</b> <i>Пламен Стоянов</i>	86	<b>IMPLEMENTATION OF SYMMETRIC CRYPTOGRAPHIC ALGORITHM AES OF AN 8-BIT PIC MICROCONTROLLER</b> <i>Plamen Stoianov</i>	9
10	<b>СРАВНЕНИЕ ВРЕМЕТО НА ИЗПЪЛНЕНИЕ НА CRUD ОПЕРАЦИИ ПРИ РЕЛАЦИОННИ И НЕРЕЛАЦИОННИ БАЗИ ОТ ДАННИ</b> <i>Диян Динев</i>	94	<b>COMPARISON OF EXECUTION TIME OF CRUD OPERATIONS IN RELATIONAL AND NON-RELATIONAL DATABASES</b> <i>Diyan Dinev</i>	10
11	<b>ИНТЕГРИРАНЕ НА АДАПТИВЕН AUTOSAR С АРХИТЕКТУРИ, ОРИЕНТИРАНИ КЪМ УСЛУГИТЕ В АВТОМОБИЛНИТЕ СИСТЕМИ</b> <i>Иван Орфей Иванов</i>	101	<b>INTEGRATING ADAPTIVE AUTOSAR WITH SERVICE-ORIENTED ARCHITECTURES IN AUTOMOTIVE SYSTEMS</b> <i>Ivan Orfey Ivanov</i>	11
12	<b>АНАЛИЗ НА СЪВРЕМЕНИ ИНТЕЛИГЕНТНИ СИСТЕМИ ЗА МОНИТОРИНГ НА СЕЛСКОСТОПАНСКИ И ПИТОМНИ ЖИВОТНИ</b> <i>Димитър В. Аврамов</i>	110	<b>ANALYSIS OF MODERN INTELLIGENT SYSTEMS FOR MONITORING AGRICULTURAL ANIMALS AND PETS</b> <i>Dimitar V. Avramov</i>	12
13	<b>РЕАЛИЗАЦИЯ НА РАЗРАБОТЕН МОДЕЛ ЗА АНАЛИЗ НА МНЕНИЯ НА БЪЛГАРСКИ ЕЗИК</b> <i>Даниела Петрова</i>	127	<b>IMPLEMENTATION OF A DEVELOPED MODEL FOR SENTIMENT ANALYSIS IN BULGARIAN LANGUAGE</b> <i>Daniela Petrova</i>	13
14	<b>СПАМ И КИБЕРСИГУРНОСТ. СПАМ-ФИЛТРИ – ТИПОВЕ И ПРИЛОЖЕНИЕ</b> <i>Стефка Попова</i>	135	<b>SPAM AND CYBERSECURITY. SPAM-FILTERS – TYPES AND USECASES</b> <i>Stefka Popova</i>	14
15	<b>ПРИЛОЖЕНИЯ НА МАШИННОТО ОБУЧЕНИЕ В УПРАВЛЕНИЕТО НА ТАКСИ БИЗНЕСА</b> <i>Тодор Тодоров</i>	141	<b>APPLICATIONS OF MACHINE LEARNING IN TAXI BUSINESS MANAGEMENT</b> <i>Todor Todorov</i>	15

# A MODEL FOR PROJECT-BASED LEARNING IN THE CONTEXT OF SMART CITIES

Nikolay Kr. Nikolov

**Abstract:** This report presents a model for project-based learning in the context of smart cities. It explains the necessity for this approach and details the steps or phases of the PBL model.

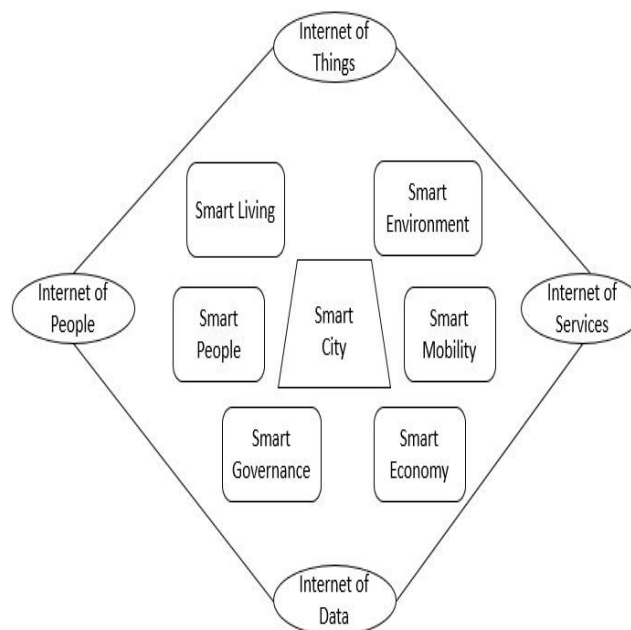
**Keywords:** advanced technology, components, IoT, PBL, smart city.

## 1. Introduction

The population growth witnessed in different parts of the world has exposed traditional cities to a new set of challenges in managing scarce resources in the most efficient manner. Urban planners have been left with the arduous task of having to plan for the expansion of these cities, some of which can no longer be expanded because of their structure, therefore, to resolve these challenges, a new paradigm has been invented in the form of smart cities. Since this concept is novel, there is a need to match it with a befitting learning model that will enable learners to develop the skills needed to operate these smart cities [1]. For this reason, Project-based learning (PBL) was devised to equip current and future urban planners and developers with the hands-on skills needed to manage and run these projects.

## 2. Smart cities defined

Smart cities are defined as urban areas that leverage data and advanced technology to enhance the delivery of urban services, promote sustainable development, and improve the quality of life for its people. Figure 1 shows one of the most widely used smart city models proposed by the US National Institute of Standards and Technology [2].



**Fig. 1.** Example of a Smart City

One key characteristic of these smart cities is that they deploy technology through interconnected systems powered by the Internet of Things (IoT) to glean real-time data that can be analyzed for actionable insights [3]. Within these cities, virtually every system is interconnected and closely monitored using IoT sensors. These IoT sensors collect real-time data which can then be analyzed for actionable insights ensuring that issues are resolved as soon as they arise. Some standard systems in smart cities include smart grids, innovative infrastructure, advanced healthcare services, and intelligent transportation systems. This new model for cities is timely as it comes at a time when urban planners grapple with challenges related to resource management, environmental sustainability, and urbanization.

### **3.3. Literature Review**

The rapid developments of smart cities necessitate the adoption of innovative educational approaches tailored to equip future professional with requisite skills to manage diverse urban systems. One such approach is Project-Based Learning (PBL) which lays emphasis on critical thinking, experiential learning and real-world problem-solving. Therefore, this current literature review is tailored to explore the application of PBL in interdisciplinary learning, information systems and higher education thereby providing insights into its feasibility, and effectiveness in the context of smart cities.

#### **3.1. Project-Based Learning in Higher Education**

PBL has been found to be effective in different higher education settings. In one study [4] is conducted an experiment to implement PBL in an Administration Theory course within an engineering curriculum at a public university in Brazil. The researcher gathered data from both the students and teachers at the end of the course using tools such as unstructured interviews, open-ended questionnaires and participant observation. Although the findings revealed that PBL increased workload both for the learners and the tutors, nonetheless, it was evaluated positively by both groups of participants. Additionally, the learners reported that the PBL approach was more dynamic and motivating compared to conventional methods. Therefore, this study has achieved the goal of underscoring the importance of PBL in higher education, most specifically in fostering engagement and motivation both of which are critical in so far as development of skills for smart city initiatives is concerned.

#### **3.2. Interdisciplinary Learning Through PBL**

The sheer scale and complexity of smart city projects necessitates the coming together of professionals from across different disciplines. In this regard in [5] proposed and tested the adoption of a PBL approach in fostering interdisciplinary learning in a smart city course. The main aim of their study was to interrogate the effectiveness and feasibility of PBL as a pedagogical framework in guiding both the teachers and the students. In order to assess the students' performance and attitudes towards the PBL approach, the researchers collected objective data via summative evaluations. From the results, it was discovered that students improved their ability to work on group projects through engagement and experience in interdisciplinary PBL. Additionally, the instructors registered their satisfaction with the learners' final project performance while at the same time, the students also expressed positive attitudes towards the approach, recognizing PBL as an effective instructional method. In their recommendations, the researchers voiced their support for the application of the approach in engineering education thereby highlighting the potential of

adopting this to learners taking courses that teach subject matters that require interdisciplinary approach such as smart cities.

### 3.3. PBL in Information System Education

PBL is also an ideal instructional approach that can be applied in information system education. According [6], the growing need for skills such as problem-solving, teamwork and managerial capabilities necessitates the adoption of instructional approaches that are more practical and hands-on. In study [7] the researchers applied a framework that aimed at monitoring, planning, execution and improving of the PBL approach. By applying this framework in the course, the researchers revealed the main benefits of PBL, such as the development of essential professional competencies and the potential for continuous improvement. This study contributes to the article by providing valuable insights into the practical implementation of PBL in information systems education, which is directly applicable to the training of professionals for smart cities.

### 4. Necessity Of Pbl Model For Smart Cities

By its very design, the nature of intelligent cities is dynamic and complex, necessitating innovative educational approaches to tackle emerging challenges. One example of an innovative educational approach that can be applied in the context of smart cities is project-based learning (PBL). This learning model is suitable because of its inherent characteristics and because it emphasizes critical thinking, experiential learning, and real-world problem-solving [8]. All the skills highlighted are critical for addressing the emerging challenges of handling novel technology.

Collaboration is another critical component of innovative city initiatives since most have many components that require various teams to handle. In this regard, PBL can be applied to foster communication and teamwork, thereby equipping students with skills and exposing them to collaborative project' experiences [9]. An example of such a project is one where engineering students work collaboratively with public policy students to design an intelligent transportation system that is technologically, environmentally, and socially sound and policy-compliant.

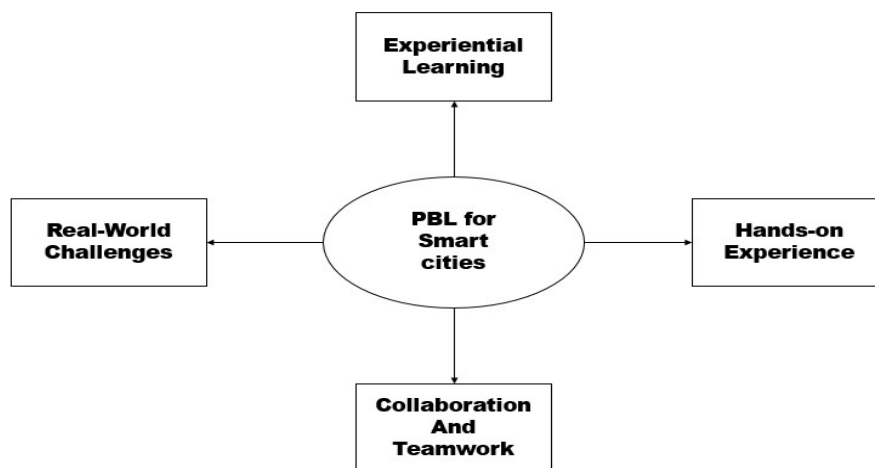


Fig. 2. Necessity of PBL Model for Smart Cities

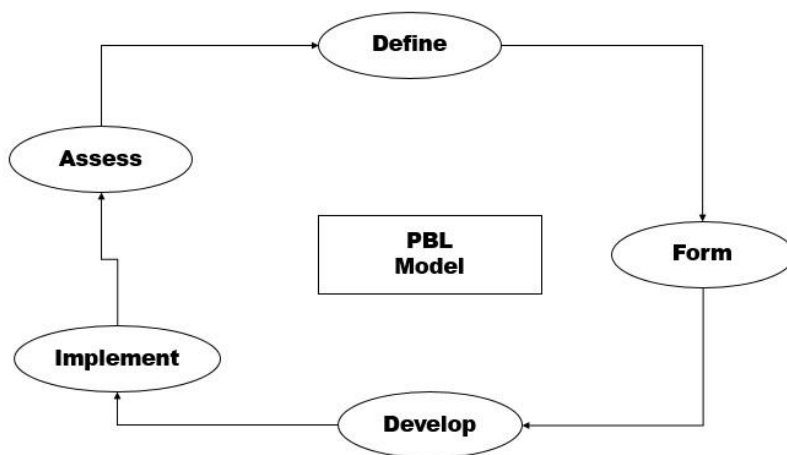
Another crucial element of PBL is technological proficiency, which is vital for successfully planning, implementing, and maintaining innovative city projects. This is so because PBL is structured to encourage learners to build hands-on experience as they interact with and use cutting-edge technologies [10]. For instance, students might engage with projects that entail developing AI algorithms for predictive learning, coding for smart sensors, or designing user-friendly apps that

would make it possible for intelligent city dwellers to interact with city services. Therefore, these practical experiences ensure that students familiarize themselves with the latest technologies while at the same time gaining an appreciation for the application of these technologies in real-world contexts.

## 5. Presenting the Model

Project-based learning (PBL) is a learning model which encourages learners to learn through real-world and meaningful project. Compared to traditional learning where rote memorization is emphasized, PBL encourages learners to actively engage and collaborate with peers in the process of learning [1]. Additionally, PBL challenges learners to explore and respond to complex questions, challenges and problems. This model of learning is ideal for modern-day learners as it allows them to build hands-on skills through working on projects that vary in complexity [11]. Therefore, through this hands-on approach to learning, learners are able to deeply engage with the subject matter while at the same time honing soft skills including problem-solving, relationship building and critical thinking which will empower them to apply their hard skills and knowledge gained in practical ways to solve real-world problems.

In the context of smart cities, the PBL model stands out as being both beneficial and necessary for a couple of reasons. Firstly, smart cities are designed in such a way that they require the professional to have a set of different skills to handle the problems. For instance, these cities are based on intricate systems and infrastructures that integrate urban planning, technology and data [12]. Therefore, PBL has the potential of equipping students with hands-on experience that they can employ to tackle the multifaceted issues that emanate from their projects. Secondly, since smart city projects require knowledge that cut across different fields, PBL is appropriate since it allows students to build relationships and collaborate with peers from different fields which mirrors what they would expect to see in real-world projects.



**Fig 3.** The PBL model in the context of smart cities

The model for PBL in the context of smart cities can be presented as a flow diagram, which shows the need for continuous improvement and adaptation. Figure 3 presents the model for PBL in the context of smart cities. The flow diagram presented below has five key components, each showing a crucial phase of the PBL cycle: define, plan, execute, review, and iterate.

### 5.1. Defining the project scope and objectives

The project objectives and scope must be clearly outlined in this first phase. This might entail identifying problems or opportunities within the brilliant city context that must be tackled,



addressed, or explored. An example can be optimizing energy consumption using IoT. Setting clear and achievable goals will elevate the chances of project success.

## **5.2. Forming Interdisciplinary Teams**

By their nature, scale, and scope, intelligent city projects often require a blend of skills from different disciplines. Forming interdisciplinary teams is crucial, and each team must ensure that it complements what the others bring to the project [13]. For instance, a smart city project team might include students from urban planning, computer science, and environmental science disciplines. This diversity is critical as it encourages adopting a holistic approach to problem-solving that mirrors real-world scenarios.

## **5.3. Developing Real-World Smart City Problems**

Students participating in innovative city projects get the opportunity to formulate solutions to real-world problems that these cities face. Solutions to these problems might call for collaborations and partnerships with local businesses or governments to address waste reduction or traffic management. By working on real-world problems, students will get opportunities to apply theory to practice and gain insights into the inner workings of smart city infrastructure and systems.

## **5.4. Implementing the Project**

This phase combines the planning, execution, and review steps of the entire project. Planning entails setting timelines, allocating resources, and defining individual and team roles. During the execution phase, students develop and test their solutions, such as deploying sensors to collect data on air quality or real-time sensors to direct traffic flow. Regular reviews will be critical in ensuring that the project stays on track and that any emerging issues get addressed promptly.

## **5.5. Assessing and Iterating**

After the project has been implemented, the following steps are to assess and iterate. Assessment is multifaceted and continuous and would entail mentor feedback, peer reviews, and self-assessment. The iterative process is critical as it affords students the opportunity to refine their solutions based on the feedback received. For instance, should a developed solution reveal inefficiency, students can improve upon this by tweaking their approach.

## **6. Conclusion**

This report has reiterated the importance of PBL in the context of smart cities, emphasizing its role in helping students hone critical skills such as collaboration, problem-solving, and technological skills. The presented model has revealed the cyclic nature of the process, which points to the importance of continuous improvement. As has been demonstrated, PBL plays a central role in preparing students to meet the demands of smart city initiatives. Therefore, future research could explore other ways emerging technologies can be integrated with external frameworks.

## **References**

- [1]. Kokotsaki, D., Menzies, V., Wiggins, A., Project-Based Learning: A Review of Literature. *Improving Schools* 19(3), 2016, pp. 267-277.
- [2]. Khatoun, R., Zeadally, S., Smart cities, *Communications of the ACM* 59 (8), 2016, pp. 46-57.

- [3]. Batty, M., Axhausen, K. W., Giannotti, F., Pozdnoukhov, A., Bazzani, A., Wachowicz, M., Portugali, Y. ,Smart cities of the future. *The European Physical Journal Special Topics*, 214, 2012, pp. 481-518.
- [4]. Mizukami, M., An experiment with PBL in higher education as appraised by the teacher and students. *International Journal of Educational Development*, 78,2020, pp. 1-12.
- [5]. Yueh, H.-P., Lin, W., Fostering interdisciplinary learning in a smart living technology course through a PBL approach. *Journal of Educational Technology & Society*, 18(1), 2015, pp. 1-10.
- [6]. Alexandre, S., Santos, J., Rodrigues, P., Applying and Managing PBL - An Experience in Information Systems Education. *Education and Information Technologies*, 23(2), 2018, pp. 585-600.
- [7]. Santos, S. C. and Rodrigues, A. “A Framework to Apply PBL in Computing Education”, FIE, Erie, Pennsylvania, 2016.
- [8]. Oliveira, Á., Campolargo, M., From smart cities to human intelligent cities. In 2015 48th Hawaii International Conference on System Sciences, 2015, pp. 2336-2344.
- [9]. Berntzen, L., Johannessen, M. R., & Florea, A., Smart Cities: Challenges and a Sensor-based Solution A research design for sensor-based intelligent city projects, 2016.
- [10]. Shin, S. Y., Kim, D., Chun, S. A., The digital divide in advanced intelligent city innovations. *Sustainability*, 13(7), 4076, 2021.
- [11]. Suradika, A., Dewi, H. I., Nasution, M. I., Project-based learning and problem-based learning models in critical and creative students. *Jurnal Pendidikan IPA Indonesia*, 12(1), 2023, pp. 153-167.
- [12]. Liu, X., Wang, X., Wright, G., Cheng, J., Li, X., Liu, R., A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and GeographicInformation System (GIS) *International Journal of Geo-Information* 6, 53, 2017.
- [13]. Villanueva-Rosales, N., Cheu, R. L., Gates, A., Rivera, N., Mondragon, O., Cabrera, S., Prochazkova, D., A collaborative, interdisciplinary initiative for an intelligent cities innovation network. In 2015 IEEE First International Smart Cities Conference (ISC2), 2015, pp. 1-2.

**For contacts:**

Nikolay Kr. Nikolov, PhD  
Department of Basic  
University of Telecommunications and Posts  
E-mail: n.nikolov@utp.bg



# DEVELOPMENT OF AN INTELLIGENT SYSTEM FOR MANAGING ENERGY CONSUMPTION IN THE HOME

Nikolay Kr. Nikolov

**Abstract:** The paper discusses the development of an intelligent energy management system that aims at reducing Energy Consumption within a residential building. The real-time data from various devices at home is harnessed and analyzed with AI, ML, and IoT to bring about energy efficiency, economization, and sustainability. The research methodology incorporates data collection, advanced analysis of collected data, and structured development of the software application. Results show a big improvement in energy savings and user convenience compared to traditional methods, thus proving that a system like this can contribute significantly toward a sustainable living environment.

**Keywords:** Artificial Intelligence, Energy Efficiency, Internet of Things, Machine Learning, Smart Home, Energy Management System, Sustainability, Data Analysis, Home Automation.

## 1. Introduction

Energy efficiency in residential buildings deals with energy savings, cost reduction, and protection of the environment. Optimizing energy use is thus found to be quite crucial in cover as global energy demands increase. Traditional ways through which energy management has been accomplished fail to capture the dynamic and variable character of household energy consumption nowadays. Efficient use of energy reduces utility bills for the householder and contributes to sustainability through reduced greenhouse gas emissions and preservation of natural resources [1]. Technology advancements in the form of smart meters and IoT devices open up new opportunities to monitor and control energy consumption in real time. This can be attained by implementing intelligent energy management systems for homeowners to make decisions on how they consume energy to use home appliances and systems more efficiently. This optimization is critical for financial savings but, more importantly, for ecological benefits, so energy efficiency will be a crucial factor in sustainable living in this century. In the present research, an integrated solution for energy use optimization in residential settings will be provided with cutting-edge technologies: artificial intelligence, machine learning, and the Internet of Things. The paper elaborates on the theoretical underpinning of intelligent energy management, describes an appropriate stringent research methodology for data collection and analysis, and outlines a detailed architecture for the software product. In addition, the work discusses the efficiency of testing and validation methods that were checked against this new approach compared to the traditional ones in energy management. By discussing these aspects, the article analyzes the broader context of discourses on sustainable living and new technological solutions that make for sustainable energy use.

## 2. Theoretical Foundations And Concept

### 2. 1. Smart Power Management

Smart power management refers to the shift of traditional energy management practices to intelligent and automated control systems aiming to achieve the optimization of energy consumption in facilities. Unlike conventional ways that rely on manual monitoring and subsequent adjustment, smart power management systems are underpinned by advanced technologies that provide real-time insights and automated responses executable with energy consumption patterns [2]. These usually incorporate intelligent meters, sensors, and other associated gadgets that measure

the amount of energy consumed, which is then analyzed for inefficiencies and possible areas of optimization to help avoid losses, reduce utility bills, and improve general efficiency of household energy use. Using intelligent power management systems will allow perfectly balanced energy demand against the supply to optimize how the resources are utilized [3]. This offers single households not just a benefit by reducing energy bills but also ensures a contribution toward broader environmental goals because it reduces the carbon footprint of residential areas.

## **2. 2. Artificial Intelligence and Machine Learning**

AI and ML are the core of any modern power management system. Artificial Intelligence is developing algorithms and systems that can perform tasks that generally require human intelligence, such as learning, reasoning, and problem-solving [4]. In energy management, AI systems are capable of acquiring vast amounts of data from different sources, recognizing the patterns, and henceforth making predictions on future energy consumption. It is further enhanced with machine learning, a subset of AI, to improve system performance over time by learning continuously from data [5]. For example, ML algorithms can use historical energy usage data to forecast peak consumption times and automate setting adjustments that can help prevent energy spikes. They also identify anomalies in energy-use patterns that can indicate inefficiencies or faults in the system. In AI and ML, intelligent power management systems will be very adaptive in optimizing energy consumption dynamically in real-time with substantial energy savings and very high efficiency

## **2. 3. Internet of Things (IoT)**

The Internet of Things provides the necessary infrastructure for interconnection among various devices and systems within a home. Hence it is one of the critical enablers in intelligent power management systems. IoT, therefore, refers to any network of physical devices embedded with sensors, software, and other technologies connecting them, which allow them to exchange data over the internet [6]. In a smart home, some IoT devices include smart thermostats, lighting systems, appliances, and energy meters, which interconnect with each other, and a central energy management system. Most importantly, it facilitates seamless collection and transmission of energy usage data in real time for monitoring and control. It can be programmed so that it turns off the lights when nobody is in one of the rooms and sets back the thermostat based on occupancy patterns, among others. With the continuous data flow and automation that IoT enables, therefore, intelligent power management systems will be highly efficient and responsive [7]. This will not only help in better energy management but also empower the homeowner to have more control and convenience while managing their energy consumption.

## **3. Research Methodology**

### **3.1. Data Collection**

Designing an intelligent energy management software application begins with a comprehensive literature review. This aims to collate many existing knowledge bases, theories, and empirical studies concerning intelligent energy management, AI, ML, and IoT technologies. The sources are primarily from leading journals, industrial reports, and authoritative books that supply information on the latest developments in this knowledge domain and best existing practices. Literature reviews show how previous research findings contribute to knowledge accumulation in the chosen area of investigation, identifying gaps in the current knowledge and laying down a foundation for making new contributions to the field [8]. This systematic literature review will also help gain data on many different energy management systems, algorithms used to realize

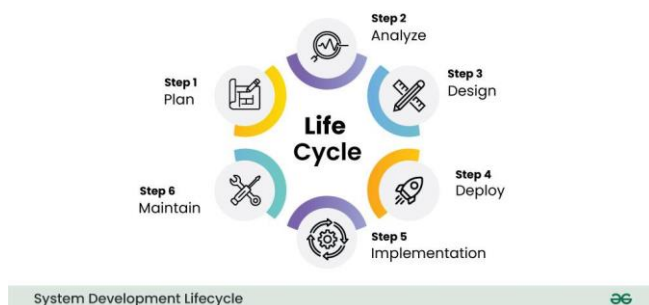
optimization, and the impact of IoT on energy consumption. This therefore also includes case study analysis and real implementation cases of similar systems to learn from to point out critical success factors.

### 3.2. Data Analysis

Literature data will be analyzed line by line for relevant insights and information to develop the software product. In the process, this will involve qualitative analysis techniques such as Thematic Analysis and Coding, identifying emergent themes and patterns related to energy consumption optimization, AI and ML algorithms, and IoT applications. Thematic analysis will gain insight into the many strategies and methodologies used or proposed with existing systems, highlighting strengths and weaknesses. Moreover, synthesizing quantitative data provided for empirical studies of such systems in terms of energy savings statistics and efficiency improvements will give a comprehensive view of the potential impact that could be brought about by intelligent energy management systems [9]. The derived insights from this analysis will, therefore, guide the design and development of the software application, ensuring that it is driven by proven strategies and effectively handles the challenges identified.

### 3.3. Software Product Development: Steps to Develop Energy Management Software Application

The development of intelligent management in software products for energy involves organized and iterative procedures to develop a robust, efficient, and user-friendly solution. The development process is divided into six major phases: requirement analysis, design, development, testing, deployment, and maintenance and updates. Each phase plays its part in getting the final product to match user requirements and perform reliably under real-world conditions. In energy management, such software would integrate well with different IoT devices, process a vast amount of data using advanced AI and ML algorithms, and provide actionable insights through an intuitive user interface. Figure 1 shows the system development life cycle.



**Fig. 1.** System Development life cycle

#### 1. Requirements Analysis

This is the first stage of requirements gathering and analysis from the end-user and the stakeholder. This comprises an understanding of the needs and problems of homeowners, an understanding of the technical specifications, and a definition of critical features and functionalities of the software. The requirement analysis creates a clear software vision and lays the foundation for further development phases.

#### 2. Design

This is where the system architecture with its blueprint of data flows and user interface will be created including integration points with IoT devices. It should underscore scalability, security, and

usability. Wireframes and prototypes of the UI are prepared to enhance the user experience and elicit feedback from stakeholders.

### 3. Development

It is the phase where software development is performed based on design specifications, app coding, integration of AI and ML algorithms for data analysis, and making the interface communicate seamlessly with IoT devices. The development process uses an agile methodology that allows iterative progress and continuous feedback.

### 4. Testing

The Software developed is thoroughly tested for functionality, performance, and security. That includes unit testing, integration testing, system testing, and User Acceptance Testing. It guarantees the software does what is expected of it, does not contain any critical bugs, and provides a clean user experience.

### 5. Deployment

After testing, the software is then deployed for use. This involves setting up the server software, configuring the necessary network settings, and ensuring all IoT devices are connected, communicating accordingly with the central system. Training is provided to the users, who are also given documentation to enable them to understand and use the software.

### 6. Maintenance and Updates

The software, upon deployment, is consistently maintained for error fixing, performance improvement, and addition of new features based on the user feedback and inclusion of technology. Continuous monitoring shall be made to ensure that the developed software remains reliable and effective in managing energy consumption.

## 4. Architecture Of The Software Product

### 4.1. System Components

The architecture of the intelligent energy management system relies on the integral parts that separately and mutually play a vital role in ensuring correctly functioning, efficiency, and user-friendliness. The critical elements include IoT devices, central processing units, data storage, optimization engines, and a user interface. It includes mainly IoT devices, such as smart meters, sensors, or connected appliances that would collect real-time data on energy consumption. This is followed by passing on the information to a central processing unit, which acts as the 'brain' of the system, controlling the flow of data and running algorithms to analyze and optimize [10].



**Fig. 2.** Key Features of an Energy Management System

Data storage solutions, often cloud-based, securely store the collected data, making it accessible for both real-time processing and historical analysis [11]. Finally, this information will

be presented intuitively and accessible through a user interface for the user to properly monitor his energy consumption and thus exercise control over the same. Figure 2 shows the key features of an energy management system.

## **4.2. Data Integration**

Data integration forms the backbone of such software architecture, which caters to gathering, consolidating, and processing information from different IoT devices. It considers the aggregation of data arriving in various formats into one format so that it is easily analyzable by the system. Middleware solutions and data integration platforms are used to handle the seamless data transfer from IoT devices to the central processing unit [12]. These platforms, hence, leverage standardized protocols and APIs to ensure device compatibility and efficient communication between them. Data integration also involves developing robust techniques in data cleaning and preprocessing for handling missing values, outliers, and noise in the data to ensure that the latter is of high quality and reliable. This integrated data provides a complete view of energy consumption patterns, hence enabling its accurate analysis and optimization [7]. By integrating into live data, the system can track energy usage to identify anomalies and adjust them timely for better energy consumption.

## **4.3. Optimization Algorithm**

At the center of the intelligent energy management system is an optimization algorithm that makes use of AI and ML to analyze data and come up with related strategies for reducing energy use. These advanced AI and ML techniques within the algorithm process patterns from the integrated dataset and predictions of progressions [4][5]. Techniques that can be used include regression analysis to model the relationship between energy consumption and various variables affecting it, clustering for grouping similar usage patterns, and time-series forecasting to project energy requirements shortly. The algorithm also incorporates reinforcement learning, through which it will be able to learn from real-time feedback and hence improve performance over continuous iterations. Comprehensively analyzing historical and real-time data, the optimization algorithm can detect inefficient operations, recommend energy-saving measures, and automate actions associated with adjusting thermostat settings, scheduling appliance usage, and lighting [13]. The algorithm's ability to adapt under changing conditions and learn from the data makes it an effective and efficient way of giving an energy management system with continual energy savings and optimized consumption.

## **4.4. User Interface**

The intelligent energy management software has a user-friendly and intuitive UI, informative for the needs that must be conveyed to the homeowner to be empowered to monitor and control his or her consumption quickly. The UI includes a dashboard displaying vital real-time data relating to energy use, cost estimation, and environmental impact [14]. It also presents predictions of the optimization algorithm with recommendations so that in light of this information, the user is better positioned to make informed decisions about their energy use. Other interactive features include customizable settings, alerts, and reports that will enrich the user's engagement and convenience to a great extent. The design is clarity- and simplicity-oriented and applies visualization in graphs, charts, and color-coded indicators to present complex data in an easily understandable visual format. This UI will enable seamlessly integrated mobile devices so house owners can track their consumption from anywhere and take control of it.

## 5. Architecture Of The Software Product

### 5.1. Tests and Validation

Testing and validation are the two essential phases in the development process of an intelligent energy management system, which must ensure that the software works under real-world conditions. The types of testing included are unit testing, integration testing, system testing, and user acceptance testing. Unit testing checks the functionality of individual components in the software; it ensures that each module is working correctly in isolation [15]. This will be followed by Integration testing, which reviews how several components can work together, thus checking if data flows smoothly around the system. System testing looks at the entire system's performance with various applied scenarios like Peak and off-peak hours, Environmental conditions, and Interaction. UAT involves real users testing the software to ensure it meets their needs and expectations. The validation phase involves benchmarking the system against predefined performance metrics, which include, but are not limited to, energy savings, response time, and user satisfaction [16]. Such rigorous tests thus identify any bugs or issues within the system and correct them toward reliability, efficiency, and user-friendliness before deployment.

**Table 1.** Test and validation

Testing Phase	Description	Objectives	Key Metrics
<b>Unit Testing</b>	Checks the functionality of individual software components	Ensure each component functions as intended.	Pass/ fail Rate Error
<b>Integration Testing</b>	Reviews how multiple components work together, ensuring data flows smoothly across the system	Validate integration and interaction between components	- Data flow accuracy - Integration error count
<b>System Testing</b>	Examines overall system performance under various scenarios like peak/off-peak hours and environmental conditions	Ensure the system functions correctly under real-world conditions	- System response time - Peak load performance
<b>User Acceptance Testing (UAT)</b>	Involves real users testing the software to ensure it meets their needs and expectations	Confirm the software meets user requirements	- User satisfaction score - Usability rating
<b>Validation</b>	Benchmarks the system against predefined performance metrics, including energy savings and user satisfaction	Ensure the system meets performance and efficiency standards	- Percentage energy savings - User satisfaction score

### 5.2. Results Analysis

The test results are analyzed to quantify the impact of systems on energy consumption and to indicate areas for improvement. Previous studies have shown intelligent energy management systems can significantly reduce energy usage by optimizing the operation of devices in households [17]. Energy use data is compared before and after the implementation of the system to quantify the saving of energy. Moreover, feedback from users concerning the usability of the system and the



effectiveness of recommendations in taking action is solicited. Statistical techniques, such as paired t-tests, are used to assess the significance of the changes in the observed energy use. The seasonal variations of household size and occupancy patterns are also accounted for in the analysis to make any variation in performance achieved by the system more complete. Through the findings obtained, researchers should be able to realize potential enhancements for the backup algorithms and user interface, therefore guaranteeing continuous improvement in efficiency over time.

### 5.3. Comparison with Traditional Methods

Compared to the traditional methods of energy management, an intelligent energy management system has enormous advantages. Traditionally, it would involve manual monitoring and adjusting, which was time-consuming to some extent and hence inefficient. On the Contrary, the intelligent system automates everything where everything happens with the provision of real-time insights and optimization strategies in the absence of constant user intervention. Traditional methods do not have the predictive capabilities of AI and ML and, hence, are unable to foresee and act on their assessments regarding future energy needs [18]. However, this intelligent system will adopt advanced algorithm-driven methods for predicting energy use and enable prognosis with proactive adjustment to achieve more efficient energy usage. Besides, traditional methods typically lack a multiple-source data integration capability that makes it hard to get an overall view of energy consumption. The data integration capabilities of this intelligent system ensure that every relevant datum is collected and analyzed to provide more significant support in decision-making [12]. Also, user satisfaction is more significant with the smart system because of its user-friendly interface and the fact that it automates most energy management features.

**Table 2.** Test and validation

Aspect	Traditional Methods	Intelligent Energy Management System
<b>Monitoring</b>	Manual	Automated
<b>Adjustment</b>	User intervention required	Real-time optimization without user intervention
<b>Predictive Capabilities</b>	None	Advanced AI and ML-driven predictions
<b>Data Integration</b>	Limited to individual sources	Confirm the software meets user requirements.
<b>Decision-Making Support</b>	Minimal, often reactive	Data-driven, proactive adjustments
<b>User Satisfaction</b>	Lower due to manual processes and inefficiencies	Higher due to automation, ease of use, and real-time insights

## 6. Advantages And Applications

### 6.1. Energy Efficiency

One of the significant benefits associated with an intelligent energy management system is allowing residential areas to optimize energy consumption levels. The technology's working principle is based on AI, ML, and the IoT, which continuously monitors and analyzes energy consumption patterns for a house to learn possible avenues of reducing wastage, hence optimizing their usage [18]. Real-time data collection from different devices enables the system to make informed decisions on when and how Energy has to be used. This would include modulation of the thermostat during peak hours or turning off appliances that are not in use. This leads to a significant amount of energy savings, usually reducing the consumption of household energy [17]. Besides,

predictive capabilities within the system ensure that energy is more efficiently used over time, adapting to changing usage patterns and environmental conditions. This net effect not only brings down energy bills for individual households but also helps to ensure a more stable and long-term viable energy grid by reducing peak demand pressures.

## **6.2. Sustainability and Ecological Footprint**

The intelligent energy management system essentially offers high sustainability and reduction in the ecological footprint of residential buildings. With an increase in the optimization of used energy, it lessens the total amount of electricity required, reducing greenhouse gas emissions and eventually lowering reliance on fossil fuels [19]. This is further supported by the system allowing very effective management and storage of excess energy from renewable sources, such as solar panels. This enables the use of clean energy, which further brings down the impact on the environment from energy use. The ability to continuously learn and adapt ensures that the efficiency gains in energy use are sustainable, making homes greener [20]. These benefits align with global efforts toward fighting climate change by promoting green living, making the intelligent energy management system a much-needed tool for environmental success.

## **6.3. Convenience for Users**

Besides energy savings and sustainability, this intelligent energy management system will benefit the end user with more convenience. Automating much of the energy management allows restriction of the needs for controls that earlier had to be done manually [21]. With a lucid, easy-to-use interface, the pattern of energy consumption will be transparently available to homeowners, enabling them to understand better how they use energy and make informed decisions. Real-time alerts and notifications keep users updated on material changes or immediate opportunities for savings, while auto-controls automatically change the settings based on occupancy and usage patterns. This kind of automation, along with insight, makes it easier to manage household energy and gives time to do something else rather than wasting it on keeping a tab on the amount of power being used [22]. Therefore, it extends into integrating the system with other devices in a smart home, creating a seamless and connected environment that best improves residents' quality of life.

## **7. Conclusion**

In summary, the benefits that an intelligent energy management system can offer are numerous, and those benefits come in the form of improved energy efficiency, reduction of ecological footprints, and increased convenience to users. Advanced technologies like AI, ML, and IoT bring real-time monitoring with predictive analysis into action for the automation of energy consumption. Such features help derive vast amounts of energy and cost savings, making homes sustainable and eco-friendly. The user-friendly interface and automation controls make energy management quite simple, hence viewing it a very convenient and efficient solution for today's modern household. In a scenario where global energy demands are rapidly increasing, intelligent energy management systems will become very instrumental in promoting green living and reducing the environmental impact resulting from residential energy use. It shows just how much potential these kinds of systems have for changing how we think about and manage energy in our homes, leading to a more efficient and sustainable future.

## References

- [1]. Smith, J., Johnson, L., Energy Efficiency in Residential Buildings. *Journal of Sustainable Energy*, 12(4), 2023, pp. 45-60.
- [2]. Himeur, Y., Alsalemi, A., Bensaali, F., Amira, A., & Al-Kababji, A., Recent trends of smart nonintrusive load monitoring in buildings: A review, open challenges, and future directions. *International Journal of Intelligent Systems*, 37(10), 2022, pp. 7124-7179.
- [3]. Natarajan, A., Krishnasamy, V., Singh, M., Occupancy detection and localization strategies for demand modulated appliance control in Internet of Things enabled home energy management system. *Renewable and Sustainable Energy Reviews*, 167, 2022, 112731.
- [4]. Sarker, I. H., AI-based modeling: techniques, applications and research issues towards automation, intelligent and smart systems. *SN Computer Science*, 3(2), 2022, p. 158.
- [5]. Alanne, K., Sierla, S., An overview of machine learning applications for smart buildings. *Sustainable Cities and Society*, 76, 2022, 103445.
- [6]. Mishra, P., Singh, G., 2023 Energy management systems in sustainable smart cities based on the internet of energy: A technical review. *Energies*, 16(19), 2023, 6903.
- [7]. Saleem, M. U., Usman, M. R., Shakir, M., Design, implementation, and deployment of an IoT based smart energy management system. *IEEE Access*, 9, 2021, pp. 59649-59664.
- [8]. Guckenbiehl, P., de Zubieta, G. C., Lindsay, N., Knowledge and innovation in start-up ventures: A systematic literature review and research agenda. *Technological Forecasting and Social Change*, 172, 2021, 121026.
- [9]. Grillone, B., Danov, S., Sumper, A., Cipriano, J., Mor, G., A review of deterministic and data-driven methods to quantify energy efficiency savings and to predict retrofitting scenarios in buildings. *Renewable and Sustainable Energy Reviews*, 131, 2020, 110027.
- [10]. Guo, X., Lang, X., Yuan, Y., Tong, L., Shen, B., Long, T., Mao, W., Energy management system for hybrid ship: Status and perspectives. *Ocean Engineering*, 310, 2024, 118638.
- [11]. Khriji, S., Benbelgacem, Y., Chéour, R., Houssaini, D. E., Kanoun, O., Design and implementation of a cloud-based event-driven architecture for real-time data processing in wireless sensor networks. *The Journal of Supercomputing*, 78(3), 2022, pp. 3374-3401.
- [12]. Alhasnawi, B. N., Jasim, B. H., Internet of Things (IoT) for smart grids: a comprehensive review. *J. Xi'an Univ. Archit*, 63, 2020, pp. 1006-7930.
- [13]. Rana, N., Latiff, M. S. A., Abdulhamid, S. I. M., Chiroma, H., Whale optimization algorithm: a systematic review of contemporary applications, modifications and developments. *Neural Computing and Applications*, 32, 2020, pp. 16245-16277.
- [14]. Sutton-Parker, J., Can analytics software measure end user computing electricity consumption?. *Clean Technologies and Environmental Policy*, 24(8), 2022, pp. 2441-2458.
- [15]. Trautsch, F., Herbold, S., Grabowski, J., Are unit and integration test definitions still valid for modern Java projects? An empirical study on open-source projects. *Journal of Systems and Software*, 159, 2020, 110421.
- [16]. Geraldi, M. S., Ghisi, E., Integrating evidence-based thermal satisfaction in energy benchmarking: A data-driven approach for a whole-building evaluation. *Energy*, 244, 2022, 123161.
- [17]. Zafar, U., Bayhan, S., Sanfilippo, A., Home energy management system concepts, configurations, and technologies for the smart grid. *IEEE access*, 8, 2020, pp. 119271-119286.
- [18]. Ahmad, T., Madonski, R., Zhang, D., Huang, C., Mujeeb, A., Data-driven probabilistic machine learning in sustainable smart energy/smart energy systems: Key developments, challenges, and future research opportunities in the context of smart grid paradigm. *Renewable and Sustainable Energy Reviews*, 160, 2022, 112128.
- [19]. Kabeyi, M. J. B., & Olanrewaju, O. A., Sustainable energy transition for renewable and low carbon grid electricity generation and supply. *Frontiers in Energy research*, 9, 2022, 743114.

- [20]. Mondejar, M.E., Avtar, R., Diaz, H.L.B., Dubey, R.K., Esteban, J., Gómez-Morales, A., Hallam, B., Mbungu, N.T., Okolo, C.C., Prasad, K.A. and She, Q., Digitalization to achieve sustainable development goals: Steps towards a Smart Green Planet. *Science of The Total Environment*, 794, 2021, p.148539.
- [21]. Mahapatra, B., & Nayyar, A., Home energy management system (HEMS): Concept, architecture, infrastructure, challenges and energy management schemes. *Energy Systems*, 13(3), 2022, pp. 643-669.
- [22]. Wang, X., Mao, X., Khodaei, H., A multi-objective home energy management system based on internet of things and optimization algorithms. *Journal of Building Engineering*, 33, 2021, 101603.

**For contacts:**

Eng. Nikolay Nikolov, PhD  
Department of Basic Training  
University of Telecommunications and Posts  
E-mail: n.nikolov@utp.bg



# INVESTIGATING THE ULTIMATE EXECUTION SPEED IN FPGA BASED GRADIENT EDGE DETECTION: TECHNOLOGY OF NON-RECURSIVE HYSTERESIS THRESHOLDING

Dimitre Kromichev

**Abstract:** For demanding applications, hysteresis thresholding is an indispensable stage in the computational flow of gradient based edge detection. Hysteresis thresholding depends entirely on recursive computations, thus rendering the FPGA based gradient edge detection unusable in the speed domain. The iterations are based on the optimal iteration stopping criteria. Hence, two facts follow: a) the minimum number of clock cycles required to execute gradient edge detection in FPGA has never been specified and explored; b) there is no objective basis for comparing the total execution times of the existing realizations. This paper proposes an organization of FPGA based hysteresis thresholding computations which guarantee the maximizing of operating frequency and the minimizing of the number of clock cycles in FPGA based gradient edge detection.

**Keywords:** hysteresis thresholding, FPGA, edge detection, execution speed, maximum operating frequency, clock cycle

## 1. Introduction

For demanding applications, hysteresis thresholding is an indispensable stage in the computational flow of gradient based edge detection. In the literature, the following hysteresis thresholding advantages are underlined: 1) better detection of edges, especially in the presence of noise [8]; 2) outstanding feature is the low threshold helping improve the edge-location accuracy [12]; 3) correctly marks the edge locations and presents nice outlines of the objects [16]; 4) the best performance under almost all test conditions due to the adjustable parameters affecting computation time and effectiveness [19]; 5) most suitable for detecting both strong and weak edges in complicated images [83][93]; 6) the ability to maintain the integrity of valid edge data, yield thin and precise edges and ensure high positioning accuracy [21]. The following disadvantages are pointed out: 1) demands much time to execute and inappropriate to be implemented in hardware in real time [13]; 2) hysteresis requires trial-and-error adjustment of two the thresholds; [6] [10].

There are various approaches to hysteresis thresholding in FPGA. A distributed Canny on Virtex 5 [1] [18], Virtex-4 [7], Virtex 2 [2] [17], and Spartan-3E [20] is proposed. It is pointed out that standard hysteresis thresholding requires computing based on the entire image statistics which prevents the processing of individual blocks independently. Heavy requirements on memory are placed and the result is large delays which prevent real-time implementation [14]. An implementation of Canny on Xilinx Spartan-6 in [3] uses double thresholding is executed by a double comparator with the data being stored in a specialized cache for the hysteresis thresholding. The cache size is  $1 \times \text{image width} \times 2$  bits. The comparison is achievable as long as a second pass is executed with the pixels read in the opposite direction. The comparison is also executed for 4 pixels in parallel. The computed data from pass # 1 of hysteresis is still 2 bits for each pixel and it is stored in an external onboard memory.  $512 \times 512$  images are used and the achieved maximum clock rate is 100 MHz. Hysteresis thresholding relies on a moving window, two comparators, FIFO buffer and OR gates [4] [15]]. In this most computationally expensive FPGA based gradient edge detection, all implementations depend entirely upon recursive calculations [9] [11]. It is generally underlined [5] that standard hysteresis operates on the whole image and its execution gets slower as the image gets larger.

This paper proposes an organization of hysteresis thresholding computations focused on accomplishing the following tasks: 1) design an organization of FPGA based hysteresis thresholding computations which is an optimal solution to the problem of speed in FPGA based gradient edge detection; 2) investigate the proposed organization of computations for maximum operating frequency  $F_{\max}$  and minimum number of clock cycles taken to secure the result  $nTclk_{\min}$ . The experimental data and draw the relevant conclusions are obtained by using the following tools: Quartus, TimeQuest Timing Analyzer, ModelSim, Scilab. The hardware description language is VHDL. The analyses and conclusions arrived at are relevant for gray scale images. Ten Intel (Altera) FPGA families (Table 1) are used to obtain the experimental data for the proposed organization of computations.

<b>Table 1. Intel (Altera) FPGA families used in the investigation</b>			
<b>Intel (Altera) FPGA family</b>	<b>Technology</b>	<b>Intel (Altera) FPGA family</b>	<b>Technology</b>
1. Cyclone	130 nm	6. Stratix	130 nm
2. Cyclone II	90 nm	7. Stratix II	90 nm
3. Cyclone III	65 nm	8. Stratix III	65 nm
4. Cyclone IV	40 nm	9. Stratix IV	40 nm
5. Cyclone V	28 nm	10. Stratix V	28 nm

## 2. Organization of computations in hysteresis thresholding

Specifics of hysteresis thresholding are focused on:

- Position in FPGA based gradients edge detection: last stage
- Pipelining is not applicable
- Entire image is stored in RAM
- Requires embedded memory proportional to the image size.

Hence, organization of computations must deal with:

- Maximum operating frequency of hysteresis thresholding  $F_{\max}(Hyst)$

$$F_{\max}(Hyst) \geq F_{\max}(G) \quad (1)$$

where

$F_{\max}(G)$  is the maximum operating frequency of Gaussian filtering

• The image statistics must not impact the minimum number of clock cycles required by hysteresis thresholding  $nTclk_{\min}(Hyst)$ . Therefore, recursive computations are excluded.

$F_{\max}(Hyst)$  depends on a single integer arithmetic: comparison. Because maximum input data width is 8 bits, it is guaranteed that (1) is satisfied under all test conditions.

Hysteresis thresholding computations are divided into two groups:

- Comparison with the high and low thresholds to define: contour pixels, possible contour pixels and non-contour pixels
- Tracking connectivity of possible contour pixels to define: contour pixels

and non-contour pixels.

Thus, connectivity is the main computational problem to be solved in hysteresis thresholding.

### 3. Tracking connectivity

Because every pixel is surrounded by 8 pixels, it follows that the optimal value of the square neighbourhood side is 3. If the input image is  $u(m,n)$ , Gaussian filter is of size  $z \times z$ , hysteresis thresholding the total number of uses of a single pixel required by hysteresis thresholding to process a single image row is presented by the expression

$$\sum nU_{\min} = [(z-1) * 2] * p \quad (2)$$

where

$\sum nU_{\min}$  is the sum of the minimum number of uses of a single pixel required by an entire row in the last edge detection module.  
 $p$  is the neighbourhood side  $p$  is a constant

$$p = const = z_{\min} = 3 \quad (3)$$

Therefore, from (2) it follows that

$$nU_{\min} = const = p = 3. \quad (4)$$

$nTclk_{\min}(Hyst)$  is a function of  $nU_{\min}$  according to expression (2) and the specifics of Simple Dual-port RAM read/write operations.

Hysteresis thresholding computations are based on:

#### 1) Phase I

In phase I defined are:

- All contour pixels according to

$$\text{if } P \geq VH \text{ then } PV = 255 \quad (5)$$

- All non-contour pixels according to

$$\text{if } PV \leq L \text{ then } PV = 0 \quad (6)$$

- All possible contour pixels according to

$$\text{if } PV \triangleright L \ \& \ PV < H \text{ then } PV = 125 \quad (7)$$

• All cases of immediate connectivity are tracked and the pixels within the range of immediate connectivity are defined as contour/non-contour pixels (5), (6) and (7) are executed simultaneously.

Because only a single pixel can be read from or written into a Dual-port RAM within a single clock cycle, the number of parallel implementations of (5), (6) and (7) depends on the number of the depth of Dual-port RAMs in which the non-maximum suppressed image is stored.

#### 2) Phase II

Phase II starts immediately after phase I is finished.

In phase II all computations are realized within the boundaries of square neighbourhood on the basis of (3) and (4) . Connectivity tracking for every pixel in case (7) is realized on the basis of individual comparison operations involving eight surrounding pixels. Thus, a pixel calculated in phase I as  $PV = 125$  has two options:

- Its value is defined as 255 and, therefore, it is a contour pixel
- Its value remains 125 and, therefore, it is not a contour pixel.

### 3) Phase III

In this phase, the final contour detected image is defined on the basis of analyzing all pixel values which are not impacted by the connectivity tracking procedure in phase II. Hence, this phase deals with defining the total numbers of 0s in the final contour detected image.

All calculations are executed simultaneously. Therefore, the number of parallel implementations depends on the number of the depth of Dual-port RAMs.

## 4. FPGA based hysteresis thresholding: functionality and RTL design

The functional model of FPGA based hysteresis thresholding is presented in Figure 1. It is based on:

- Dual-port RAMs to store the entire image before the start of hysteresis thresholding and after hysteresis thresholding computations are finished
- Registers to store pixels read from Dual-port RAM
- Registers to store the values of  $H$  and  $L$  during the entire process of hysteresis thresholding
- A register to store the signal coming from non-maximum suppression module and controlling the start of hysteresis thresholding operations
- Comparisons units

The basic functionality defined by the model in Figure 1 uses the following denotations:

- RAM: Dual-port RAM storing the entire image. The non-maximum suppressed pixel values are read from and the processed pixel values are stored in the same RAMRegRow1, RegRowN: 8 bit registers for each of the image rows stored in Dual-port RAM
- Comp: comparison operations required by the computations in phase I, phase II and phase III
- RegTemp1, RegTempN: 8 bit registers used to store values which must be used in tracking connectivity

*Start of hysteresis thresholding*

Clock cycle #1

RAM -> RegRow1

.

.

-> RegRowN

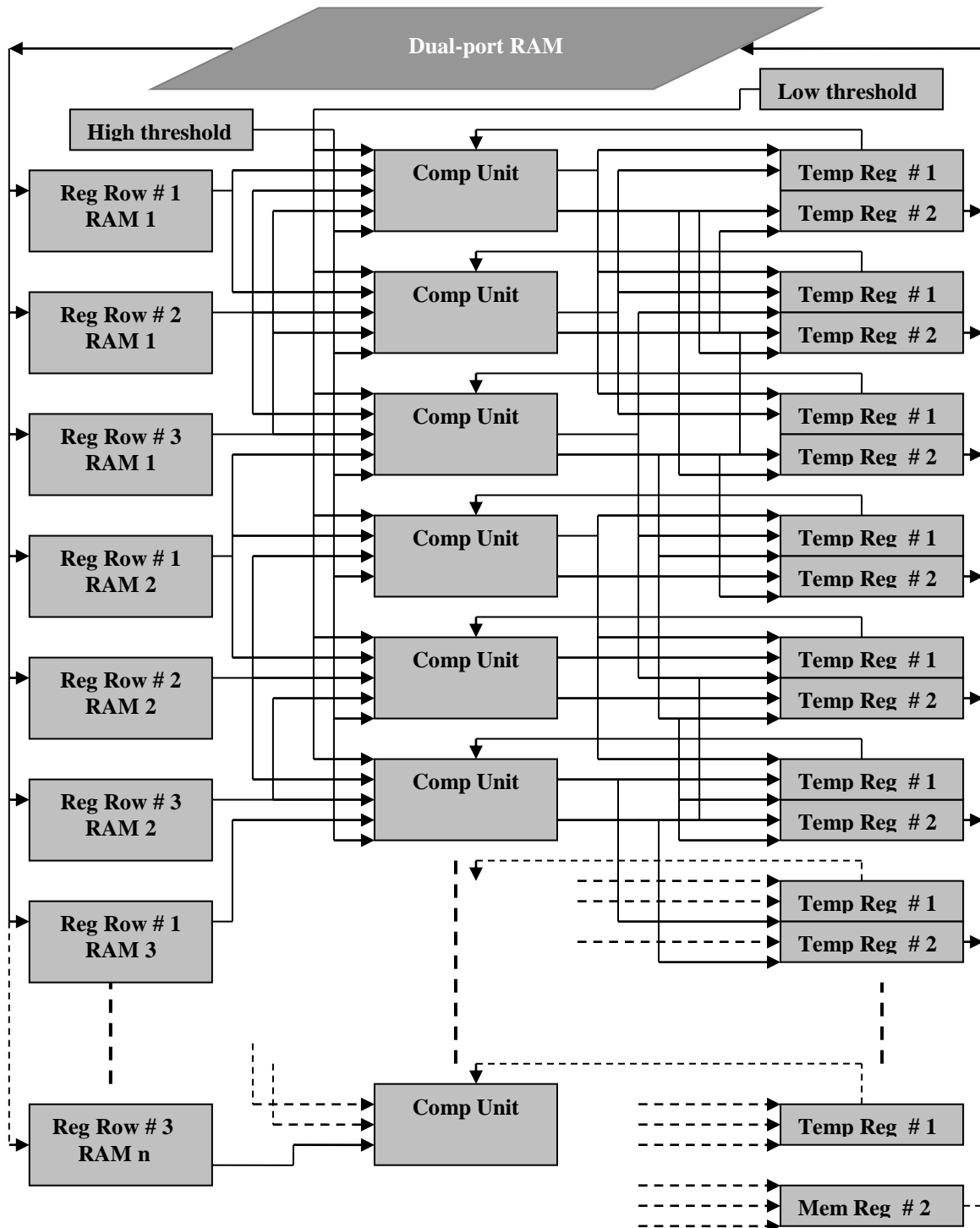
Clock cycle #2

RegRow1 -> Comp -> RegTemp1  
RegTemp1

RegRowN -> Comp -> RegTempN  
RegTempN



*Clock cycle #3*  
*RegTemp1 -> RAM*  
 .  
 .  
*RegTempN -> RAM*



**Fig. 1.** Functional model of hysteresis thresholding

*Clock cycle #4*

RAM -> RegRow1  
 -> RegRowN

Clock cycle #5

RegRow1 -> Comp -> RegTemp1  
 RegTemp1 -> -> TempMem1  
 RegRowN -> Comp -> RegTempN  
 RegTempN -> -> RegTempN

Clock cycle # $[n - (z - 1) * 2] * (p - 1)$

RegTemp1 -> RAM

.

RegTempN -> RAM.

RESULT:

For a non-maximum suppressed image row stored in a separate Dual-port RAM

$$nTclk_{min}(Hyst) = [n - (z - 1) * 2] * (p - 1).$$

The RTL design of hysteresis thresholding is shown below (Figure 2).

All hysteresis thresholding computations are executed within MEM\_HYST section of the RTL design in Figure 4.17 MEM\_HYST has eight inputs:

- *reset\_h*: the reset for sequential logic
- *clk\_h*: the system clock for sequential logic
- *ena\_h*: enable for sequential logic
- *MAX\_in*: the non-maximum suppressed pixel value set at Dual-port RAM input
- *ThH\_in*: the high threshold value used in MEM\_HYST
- *ThL\_in*: the low threshold value used in MEM\_HYST
- *start\_hyst\_in*: an internal signal which defines the start of hysteresis thresholding computations

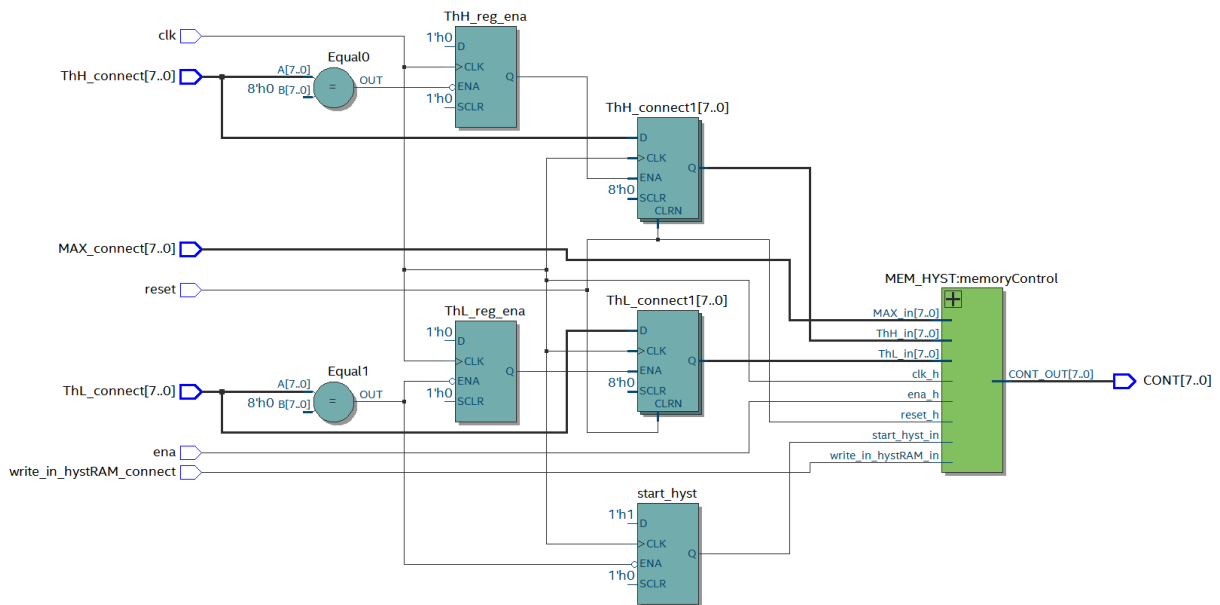


Fig. 2. RTL design of hysteresis thresholding (Source: Intel (Altera) Quartus RTL Viewer)

- *write\_in\_hystRAM\_in*: control signal from the previous gradient based edge detection module defining the start of writing the non-maximum suppressed pixels into Dual-port RAM.

In the RTL design *MEM\_HYST* has a single output: *CONT*. This port used to transfer the contour detected image.

The two registers – *ThH\_teg\_ena* and *ThL\_teg\_ena* and are of critical importance because they guarantee that the values of high and low thresholds will remain constant during the entire process of hysteresis thresholding. All hysteresis thresholding computations are realized within *MEM\_HYST*.

Resource utilization of hysteresis thresholding for image of size 148x148 is presented in Table 2.

FPGA family	Resource utilization of of hysteresis thresholding for image of size 148x148				
	Logic utilization (in LEs/ALUTs/ALMs)	Total registers	Total memory bits	Total DSP blocks	Embedded multiplier 9 bit elements
<b>Cyclone</b>	282 (LEs)	-	286720	-	-
<b>Cyclone II</b>	253 (LEs)	182	286720	-	0
<b>Cyclone III</b>	228 (LEs)	167	286720	-	0
<b>Cyclone IV</b>	228 (LEs)	167	286720	-	0
<b>Cyclone V</b>	84 (ALMs)	167	286720	0	-
<b>Stratix</b>	234 (LEs)	194	286720	0	-
<b>Stratix II</b>	181 (ALUTs)	178	286720	0	-
<b>Stratix III</b>	169 (ALUTs)	161	286720	0	-
<b>Stratix IV</b>	169 (ALUTs)	161	286720	0	-
<b>Stratix V</b>	82 (ALMs)	161	286720	0	-

## 5. Exploring the speed of hysteresis thresholding

Exploration methodology:

- The proposed organization of computations in hysteresis thresholding is compared with the recursive approach
- Five real life images of size 148x148 are processed using Scilab Image Processing Toolbox and Gaussian filter is of size 5x5. The size of the images is chosen to be 148x148 because of the limited memory resources in Cyclone family. After executing Gaussian filtering and Sobel filtering, the non-maximum suppressed image matrix is of size 140x140. Each of these five image matrices is stored in Dual-port RAMs
- A special counter module is designed. The task is to count all rising clocks in the complete hysteresis thresholding computations
- The image is stored in four Dual-port RAMs
- The following tools are used to obtain the experimental data and draw the relevant conclusions: Scilab Image Processing Toolbox, Quartus, TimeQuest Timing Analyzer, ModelSim
- The precision of detected contours is ascertained and the relevant conclusions are drawn using the developed methodology.

The achieved results are shown in Table 3 and Table 4.

## 6. Analysis of results

**Table 3.**  $F_{\max}(Hyst)$  and  $nTclk_{\min}(Hyst)$  of the designed organization of hysteresis thresholding computations for image of size 148x148

FPGA family	Speed of the designed organization of hysteresis thresholding computations for image of size 148x148					
	$F_{\max}(Hyst)$ (in MHz)	$nTclk_{\min}(Hyst)$				
		Non-maximum suppressed image matrix				
		#1	#2	#3	#4	#5
Cyclone	232	88320	88320	88320	88320	88320
Cyclone II	319	88320	88320	88320	88320	88320
Cyclone III	367	88320	88320	88320	88320	88320
Cyclone iV	370	88320	88320	88320	88320	88320
Cyclone V	388	88320	88320	88320	88320	88320
Stratix	331	88320	88320	88320	88320	88320
Stratix II	464	88320	88320	88320	88320	88320
Stratix III	589	88320	88320	88320	88320	88320
Stratix IV	637	88320	88320	88320	88320	88320
Stratix V	685	88320	88320	88320	88320	88320

**Table 4.**  $F_{\max}(Hyst\_recursive)$  and  $nTclk_{\min}(Hyst\_recursive)$  of recursive hysteresis thresholding for image of size 148x148

FPGA family	Speed of recursive hysteresis thresholding computations for image of size 148x148					
	$F_{\max}(Hyst\_recursive)$ (in MHz)	$nTclk_{\min}(Hyst\_recursive)$				
		Non-maximum suppressed image matrix				
		#1	#2	#3	#4	#5
Cyclone	232	466786	512452	482440	508616	520084
Cyclone II	319	466786	512452	482440	508616	520084
Cyclone III	367	466786	512452	482440	508616	520084
Cyclone iV	370	466786	512452	482440	508616	520084
Cyclone V	388	466786	512452	482440	508616	520084
Stratix	331	466786	512452	482440	508616	520084
Stratix II	464	466786	512452	482440	508616	520084
Stratix III	589	466786	512452	482440	508616	520084
Stratix IV	637	466786	512452	482440	508616	520084
Stratix V	685	466786	512452	482440	508616	520084

The data in Table 3 and Table 4 shows that  $F_{\max}(Hyst)$  and  $F_{\max}(Hyst\_recursive)$  satisfy (1).

The data in Table 3 shows that  $nTclk_{\min}(Hyst)$  is a constant. Therefore,  $nTclk_{\min}(Hyst)$  does not depend on the non-maximum suppressed image statistics.

The data in Table 4 shows that  $nTclk_{\min}(Hyst\_recursive)$  is a variable. Therefore,  $nTclk_{\min}(Hyst\_recursive)$  is impacted by the non-maximum suppressed image statistics. In terms of  $nTclk_{\min}$ , the data in Table 3 and Table 4 shows that the designed organization of hysteresis thresholding computations is from 80.08% to 83.1% faster than recursive hysteresis thresholding.

Contour precision achieved by the proposed organization of hysteresis thresholding computations and by recursive hysteresis thresholding is the same for all test images.

## 7. Conclusions

Experimentally investigated and analyzed in this paper is an approach to FPGA based hysteresis thresholding be used in gradient based edge detection. The upper speed limit of the proposed organization of hysteresis thresholding computations is investigated in ten Intel (Altera) FPGA families for maximum operating frequency and minimum number of clock cycles. It is ascertained that: 1) the number of clock cycles in the proposed organization of hysteresis thresholding computations does not depend on the non-maximum suppressed image statistics; 2) the designed organization of hysteresis thresholding computations is from 80.08% to 83.1% faster than recursive hysteresis thresholding. The achieved contour precision of the proposed organization of hysteresis thresholding is the same as the precision obtained by using recursive hysteresis thresholding computations for all test images.

## References

- [1]. Alka Singh, Kanika Jindal, Realization of Image Processing Algorithm based on FPGA, International Journal of Recent Technology and Engineering (IJRTE), Volume-8, 2019, pp. 80-84
- [2]. Avinash G. Mahalle 2017 A Review on FPGA Implementation of Edge Detection Algorithms International Journal of Science and Research (IJSR) Vol. 6(1), pp. 1104 - 1107
- [3]. Chandrashekar N.S K.R. Nataraj, A Review on FPGA Implementation of Distributed Canny Edge Detector, Proc. of Int. Conf. on Current Trends in Eng., Science and Technology, ICCTEST 2017, pp. 394-399
- [4]. Diya Chudasama, Tanvi Patel, Shubham Joshi and Ghanshyam I. Prajapati, Survey on Various Edge Detection Techniques on Noisy Images, International Journal of Engineering Research & Technology, Vol. 3 (10), 2014, pp. 705-709
- [5]. D. Sangeetha P. Deepa, FPGA implementation of cost-effective robust Canny edge detection algorithm, Journal of Real-Time Image Processing (JRTIP), Vol. 12, (1), 2016, pp. 1-14
- [6]. Heena S. Shaikh, Narayan V. Marathe (2017) Implementation Of Distributed Canny Edge Detection Technique, International Research Journal of Engineering and Technology (IRJET) Volume: 04 (05), pp. 2926-2928
- [7]. K. L. Amruta and V. G. Sangam (2016) Canny edge detection Algorithm, International Journal of Advanced Research in Electronics and Communication Engineering, vol. 5 (5), pp. 1288-1292
- [8]. K. Supraya, Hardware Implementation Of Canny Edge Detection Algorithm With FPGA, Journal of Engineering Science and Technology, Vol. 12, No. 9, 2017, pp. 2536-2550
- [9]. Lakshamma K M., Chandana B.R, RTL Design and FPGA Implementation of Canny Edge Detector with Real Time Threshold Adjustment Capability, International Journal of Science and Research (IJSR), Vol. 4, (4), 2015, pp. 1731-1733

- [10]. M. A. Raja, C. Manju, B. Aruna Devi, S. Kalaivani, FPGA Implementation of Novel Real Time Distributed Canny Edge Detection Algorithm, Asian Journal of Research in Social Sciences and Humanities, Vol. 7 (5), 2016, pp.813-835
- [11]. Meghanand Kumar, Sejal Singh (2020) Hardware and Software Implementation of Canny Edge Detection Algorithm, Journal of Information and Computational Science, Volume 10 (4), pp. 415 – 422
- [12]. Mohammed Sabah, Dr. R. Sundaraguru, (2018) A Survey on Improved Canny-Edge Detection Algorithm, Perspectives in Communication, Embedded-Systems and Signal-Processing, Vol. 2 (4), pp. 75-81
- [13]. Pramod Kumar S, Narendra T.V, Vinay N.A, Short Hand Recognition using Canny Edge Detector, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 7, (5), 2017, pp. 901-907
- [14]. Raja M. A., Manju C., Devi B. Aruna, Kalaivani S. (2016) FPGA Implementation of Novel Real Time Distributed Canny Edge Detection Algorithm Asian Journal of Research in Social Sciences and Humanities 2016, Volume 6 (5), pp. 813 – 835
- [15]. R. Jayarani, Edge Detection using Distributed CannyAlgorithm and Implementation in FPGA, International Journal of Research in Engineering, Science and Management Volume-2 (5), 2019, pp. 1155-1150
- [16]. Vandana Thind, Gaurav Verma, Nisha Pandey (2017) Power and Time Delay Analysis of Simple Comparator Implemented On Different Type of FPGA, Journal of Engineering and Technology, Vol.3 (1), pp. 1-7
- [17]. Vinayaka pallakki V., pushpalatha K. N. Review, Paper on Distributed Canny Edge Algorithm Implementation on FPGA International Research Journal of Engineering and Technology (IRJET), Vol.6(1), 2019, pp. 1081 – 1088
- [18]. V. Raghavendra and L. Shrinivasan, Time Efficient Design and FPGA Implementation of Distributed Canny Edge Detector Algorithm, IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2018, pp. 2135-2139
- [19]. Yingke Feng Jinmin Zhangand Siming Wang A new edge detection algorithm based on Canny idea, 2nd International Conference on Materials Science, Resource and Environmental Engineering (MSREE 2017), 2017, pp. 1 – 7
- [20]. Yu Chen, Caixia Deng, Xiaxia Chen, An Improved Canny Edge Detection Algorithm, International Journal of Hybrid Information Technology, Vol.8 (10), 2015, pp.359-370
- [21]. Y. Jang, J. Mun, Y. Nam and J. Kim (2019) Novel Hysteresis Thresholding FPGA Architecture for Accurate Canny Edge Map, 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), pp. 1-3

**For contacts:**

Dimitre Zhivkov Kromichev, PhD  
 Department of Marketing and International Economic Relations  
 University of Plovdiv  
 e-mail: [dkromichev@yahoo.com](mailto:dkromichev@yahoo.com)

# TECHNOLOGY OF OPTIMIZING THE SPEED IN FPGA BASED GRADIENT EDGE DETECTION: PARALLEL EDGE THINNING AND THRESHOLD COMPUTATIONS

Dimitre Kromichev

**Abstract:** Two important criteria for precision in edge detection are: a single pixel thin contours and continuous contours. Satisfying these criteria by applying the traditional (John Canny's) approach requires the addressing of specific problems in FPGA based gradient edge detection. Firstly, the traditional mathematics of edge thinning leads to false negatives in a number of cases for various image statistics. This problem was addressed, thoroughly analyzed and resolved by proposing a correction to the non-maximum suppression mathematics. Secondly, the threshold calculation in a sequential fashion by using the entire image statistics is incompatible with the concept of speed and the use of FPGA based gradient edge detection for demanding applications in real time. The latter is dealt with and resolved in this paper by proposing a non-maximum suppression with simultaneous dynamic threshold computation. The organization of computations is investigated for its upper speed limit in ten Intel (Altera) FPGA families.

**Keywords:** FPGA, gradient edge detection, non-maximum suppression, dynamic threshold computation, maximum operating frequency, clock cycles

## 1. Introduction

There is a number of approaches to the problem of threshold computation in gradient edge detection. They can be summarized as: 1) within-class variance method, 2) customized threshold function; 3) self-adaptive threshold computation; 4) connected component analysis algorithm, 5) genetic algorithm, 6) adaptive threshold computation by doing a differential operation on amplitude gradient histogram, 6) adaptive threshold selection method, 7) adaptive dual-threshold detection method; 8) self-adaptive threshold; 9) type-2 fuzzy set-based technique; 10) S-membership function [16]. All of them are executed as a separate stage after the entire image has been non-maximum suppressed [11][12] [13]. The most widely applied is the within-class variance method used for adaptive thresholding based on calculating: 1) the gray level gradient histogram; 2) the probability of background and object; 3) the maximum between-cluster variance of different regions; 4) the threshold with maximum between-cluster variance  $\sigma$ ; 5) the low threshold as the high threshold value multiplied by 0.4 or 0.5 [2][10]. Automatic selection of the thresholds relies on gradient histogram concavity analysis. High threshold is selected between the largest peak and the second largest peak by using a non-uniform quantizer [17]. The following disadvantage is pointed out: use of gradient calculation to define the angle of suppression [18]. In FPGA implementations there are several general approaches. In distributed Canny [14][15] an intermediate classification threshold is calculated based on a set of pixels with gradient magnitudes larger than a defined value; then the 64-bin uniform discrete histogram is used for the high threshold calculation. Low threshold is 40% of the high threshold. 128x128, 256x256 and 512x512 images are used. Maximum reported clock frequency is 100 MHz. No counting methodology is provided, and, therefore the total number of clock cycles for executing the distributed Canny is not specified. When derivative and smoothing operations are combined into a single mask high and low thresholds are fixed to constant values [9]. In block level implementation standard non-maximum suppression is used, and then uniformly quantized gradient magnitude histograms are computed on overlapped blocks. Adaptive thresholding is based on block classification [19].

Although the concept of selecting the largest value pixel as a technique to thin the detected contour had been in circulation for quite some time it is the John Canny's paper of 1986 [1] that defined this approach as a required stage in gradient edge detection. The approach in [1], referred to as traditional, presents two capital problems to the FPGA implementation of gradient edge detection: 1) the mathematics of edge thinning given in [1] leads to 50% of false negatives in a number of cases and various image statistics. This problem was addressed, thoroughly analyzed and resolved in [3] by proposing a correction to the non-maximum suppression mathematics; 2) the threshold calculation in a sequential fashion by using the entire image statistics is incompatible with the concept of speed and the use of FPGA based gradient edge detectoin for demanding applications in real time. The latter is addressed and resolved in this paper by proposing a non-maximum suppression technology with simultaneous dynamic threshold computation. The upper speed limit of the proposed organization of computations is experimentally investigated. The analyses and conclusions are relevant for gray scale images. Ten Intel (Altera) FPGA families are employed in the investigations. The follwing tools are used to obtain the experimental data and draw the relevant conclusions: Quartus, TimeQuest Timing Analyzer, ModelSim, Scilab. The hardware description language is VHDL.

## 2. Dynamic threshold computation

The non-maximum suppression and dynamic threshold computation must deal with:

- Maximum operating frequency of non-maximum suppression

$F_{\max}(NMS)$

$$F_{\max}(NMS) \geq F_{\max}(G) \quad (1)$$

where

$F_{\max}(G)$

is the maximum operating frequency of Gaussian filtering

- Minimum number of clock cycles required by non-maximum suppression

$nTclk_{\min}(NMS)$

$$nTclk_{\min}(NMS) \leq nTclk_{\min}(M) / nTclk_{\min}(D) \quad (2)$$

- Pipelining efficiency

$$nTclk(pausePixel) = Tclk_{\min Tclk}(NMS) - nTclk_{\min}(M) / nTclk_{\min}(D)$$

$$nTclk(pausePixeNMS)_{\max} = 3$$

$$nTclk(pausePixeNMS)_{\min} = 2 \quad (3)$$

where

$nTclk(pausePixeNMS)$

is the pause between the processing of two consecutive image pixels

in non-maximum suppression measured in clock cycles,

$nTclk(pausePixeNMS)_{\max}$  is the maximum value of  $nTclk(pausePixeNMS)$ ,

$nTclk(pausePixeNMS)_{\min}$  is the minimum value of  $nTclk(pausePixeNMS)$ .

$$nTclk(pauseTotaNMS) = \{m - (z - 1) * 2\} * [n - (z - 1) * 2] * nTclk(pausePixeNMS) - 1 \quad (4)$$

where

$nTclk(pauseTotaNMS)$

is the accumulated pause in non-maximum suppression calculated on the basis of all image pixels, and measured in clock cycles,



$m/n$

is the number of rows/columns in the input image  $u(m,n)$ .

The goal here is to compute the accurate thresholds:

- in a dynamic fashion
- simultaneously with the non-maximum suppression.

The problem of calculating the minimum steps between two successive high threshold values and the ratios between high threshold value and low threshold value is solved in Table 1.

<b>Table 1.</b> High and low thresholds' selection options							
<b>Parameters</b>	<b>Values</b>						<b>Sobel, gradient magnitude, gradient direction</b>
	<b>Minimum step between two successive high threshold values</b>						
	<b>5</b>		<b>7</b>		<b>10</b>		
	<b>Ratio between high and low thresholds</b> 2, 2.2, 2.5, 2.8, 3	<b>Ratios between high and low thresholds</b> 2, 25, 3	<b>Ratios between high and low thresholds</b> 2, 2.2, 2.5, 2.8, 3	<b>Ratios between high and low thresholds</b> 2, 25, 3	<b>Ratios between high and low thresholds</b> 2, 2.2, 2.5, 2.8, 3	<b>Ratios between high and low thresholds</b> 2, 2.5, 3	
<b>Number of high thresholds</b>	50	50	36	36	25	25	-
<b>Value range of high threshold</b>	[5,250]	[5,250]	[7,252]	[7,252]	[10,250]	[10,250]	
<b>Calculated high and low threshold pairs</b>	250	150	180	108	125	75	
<b>Combinations b/n approximations</b>	-	-	-	-	-	-	16
<b>Combinations b/n exact math and approximations</b>	-	-	-	-	-	-	44

On that basis, defined is a closed interval of integers  $TH$  such that

$$[(N_{\max}+1)-N_{\min}=RTH] \& [\frac{RTH}{7}=0] \& [N_{\max}+N_{\min}<2^8-1] \quad (5)$$

where

$N_{\max}$  is the rightmost value in  $TH$ ,

$N_{\min}$  is the leftmost value in  $TH$ ,

$RTH$  is the result of  $(N_{\max}+1)-N_{\min}$ .

$TH$  is divided into subintervals  $SUB\_TH$ 's in which the ratio between the rightmost value  $SUB\_N_{\max}$  and the leftmost value  $SUB\_N_{\min}$  is defined by the equation

$$(SUB\_N_{\max}+1) - SUB\_N_{\min} = 6 . \quad (6)$$

Each  $SUB\_TH$  subinterval is associated with a reference table value  $RTV$  which is defined as

$$RTV = (SUB\_N_{\max} - 3) = SUB\_N_{\min} + 3 \quad (7)$$

Defined is a set of table values – they are integers calculated by dividing the values in the interval [5, 250] by these five values {2,2.2,2.5,2.8,3}.

Defined is a set of counters  $COUNT\_SUB\_TH$ . Each counter is associated with a concrete  $SUB\_TH$ . Starting with non-maximum suppressed image pixel #1, each pixel value  $PV$  is checked for

$$SUB\_N_{\max} \geq PV \geq SUB\_N_{\min} \quad (8)$$

If (8) is true, then

$$COUNT\_SUB\_TH = COUNT\_SUB\_TH + 1 \quad (9)$$

When the entire image is non-maximum suppressed the values in the counters are compared to select the two largest among them.

The values in the counters depend on image size and image statistics. Therefore, to satisfy (1) the input data widths of comparison must be limited to 17 bits. This is guaranteed by using the following dependence:

if  $CV1 > CV2$  or  $CV1 = CV2$  or  $CV1 < CV2$  then

$$\frac{CV1}{2^n} > \frac{CV2}{2^n} \text{ or } \frac{CV1}{2^n} = \frac{CV2}{2^n} \text{ or } \frac{CV1}{2^n} < \frac{CV2}{2^n},$$

$$CV1 \in N, CV2 \in N, n \in N, n \geq 1. \quad (10)$$

Hence, for all  $COUNT\_SUB\_TH$  the operation

$$COUNT\_SUB\_TH \gg n \quad (11)$$

is executed in parallel before starting the comparison operation.

Selected are the two  $SUB\_TH$ 's and their  $RTV$ 's which correspond to these two largest values. Calculation of high threshold is based on the smaller of the two:  $SUB\_TH_{smaller}$  and its  $RTV_{smaller}$ .

For each  $SUB\_TH_{smaller}$  and its  $RTV_{smaller}$  defined is a value  $H$  by

$$H = RTV_{smaller} \cdot \frac{RTV_{smaller}}{C_H} \quad (12)$$

where

$C_H$  is a number defining the ratio between the values of  $RTV_{smaller}$  and  $H$ ;  $C_H \in \{3,4\}$ .

$H$  is the high threshold.

The high threshold is associated with a set of table values which are precalculated by using a divisor from {2,2.2,2.5,2.8,3}. The selected table value  $L$  is the low threshold.

## 2. FPGA based non-maximum suppression with dynamic hreshold computation: functionality and RTL design

The functional model of non-maximum suppression with dynamic hreshold computation is in Figure 1:

### Start.of non-maximum suppression with dynamic threshold computation

#### Clock cycle #1

Magn Regs -> Comp P II 0 – P - PI 0 -> True/False -> RAM

Direct Reg -> P II 45 - P - PI 45  
P II 90 – P - PI 90  
P II 135 – P - PI 135

-> True -> Thresh Reg

CountNMS -> Count NMS +1

#### Clock cycle #2

Magn Regs -> Comp P II 0 – P - PI 0 -> True/False -> RAM

Direct Reg -> P II 45 – P - PI 45  
P II 90 – P - PI 90  
P II 135 - P - PI 135

-> True -> Thresh Reg

CountNMS -> Count NMS +1

Thresh Reg -> Intervals -> True -> Count

=

=

Clock cycle # $[m - (z - 1) * 2] * [n - (z - 1) * 2] - 39$  through # $[m - (z - 1) * 2] * [n - (z - 1) * 2] - 15$

CountRTV Count36 ≥ Count35 -> True -> Count Reg

Count Reg -> CountRTV Count35 ≥ Count34 -> True -> Count Reg

Count Reg -> CountRTV -> ..

=

=

Clock cycle # $[m - (z - 1) * 2] * [n - (z - 1) * 2] - 2$

RTV -> Sub -> RTV >> 4 -> Count Reg

Clock cycle # $[m - (z - 1) * 2] * [n - (z - 1) * 2] - 1$

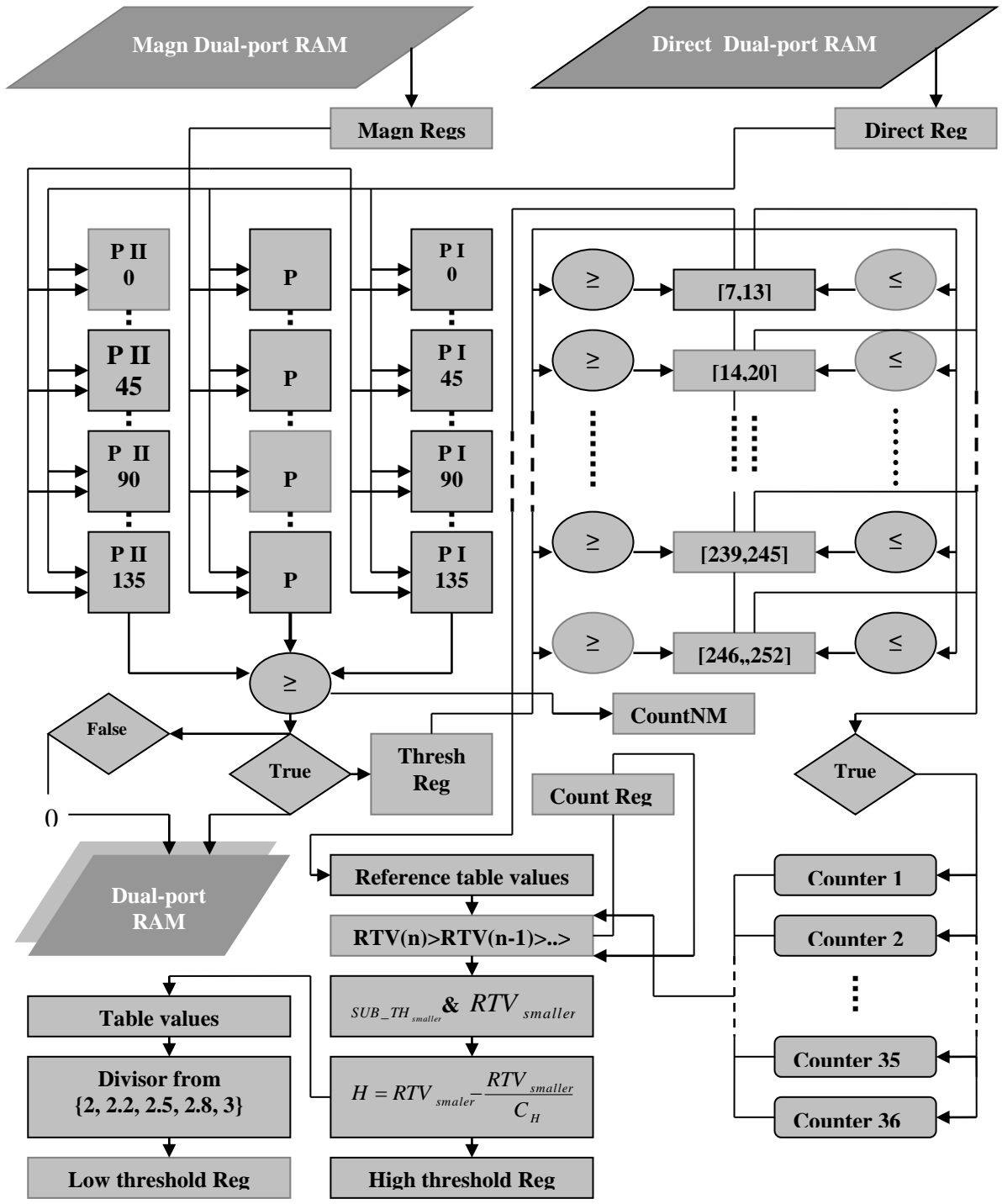
Count Reg -> Sub -> RTV >> 8 -> High threshold Reg

Clock cycle # $[m - (z - 1) * 2] * [n - (z - 1) * 2]$

High threshold Reg -> Table -> Low threshold Reg.

#### RESULT:

When the last image pixel is non-maximum suppressed the high and low thresholds are dynamically computed. Therefore, hysteresis thresholding can start immediately after the last image pixel is non-maximum suppressed. This contributes to the optimal value of  $nTclk_{min}$  in FPGA based gradient edge detectoin.

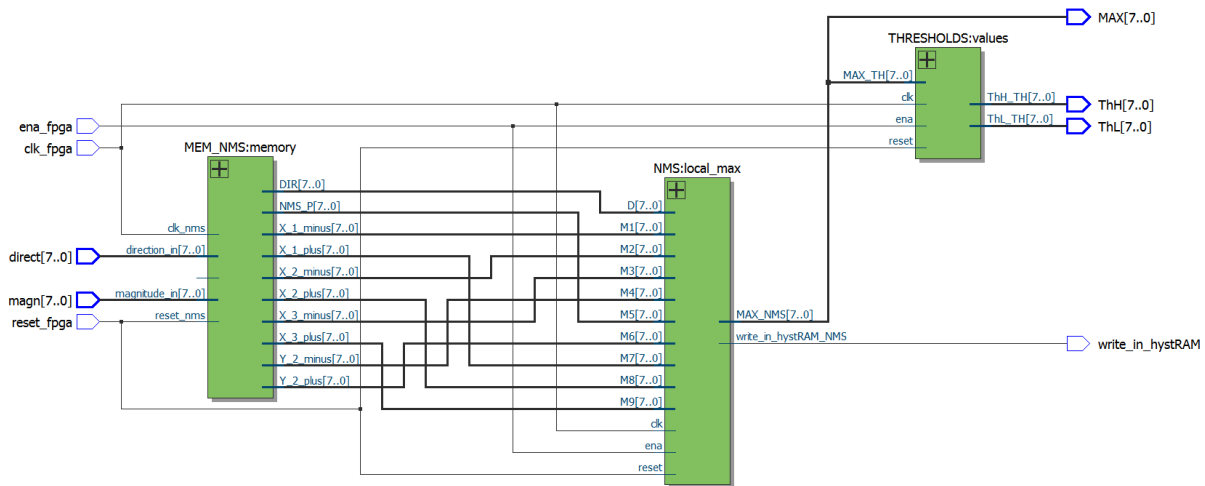


**Fig. 1.** Functional model of non-maximum suppression with dynamic threshold computation

This functionality is implemented in VHDL. The RTL design of non-maximum suppression and dynamic threshold computations is presented below (Figure 2). Resource utilization of non-maximum suppression with dynamic threshold computations and memory section is shown in Table 2.

### 3. Exploring the speed of non-maximum suppression with dynamic computation of high and low thresholds

Exploration methodology:



**Fig. 2.** RTL design of non-maximum suppression and dynamic threshold computations (Source: Intel (Altera) Quartus RTL Viewer)

<b>Table 2.</b> Resource utilization of non-maximum suppression, dynamic threshold computations and memory control section					
<b>FPGA family</b>	<b>Resource utilization of non-maximum suppression, dynamic threshold computations and memory control section</b>				
	<b>Logic utilization (in LEs/ALUTs/ALMs)</b>	<b>Total registers</b>	<b>Total memory bits</b>	<b>Total DSP blocks</b>	<b>Embedded multiplier 9 bit elements</b>
<b>Cyclone</b>	2809 (LEs)	-	65536	-	-
<b>Cyclone II</b>	2741 (LEs)	1378	65536	-	0
<b>Cyclone III</b>	2672 (LEs)	1283	65536	-	0
<b>Cyclone IV</b>	2672 (LEs)	1283	65536	-	0
<b>Cyclone V</b>	1022 (ALMs)	1308	65536	0	-
<b>Stratix</b>	2440 (LEs)	1422	65536	0	-
<b>Stratix II</b>	2215 (ALUTs)	1341	65536	0	-
<b>Stratix III</b>	2045 (ALUTs)	1279	65536	0	-
<b>Stratix IV</b>	2045 (ALUTs)	1279	65536	0	-
<b>Stratix V</b>	1012 (ALMs)	1303	65536	0	-

- All available approaches to calculation of high and low thresholds are characterized by the following two facts:

- A) Insert a whole new module in FPGA based gradient edge detection, thus changing the original concept of Canny architecture;
- B) FPGA based gradient edge detection computations stop executing until the entire non-maximum suppressed image is processed and then high and low thresholds are computed.

On that basis,  $F_{\max}(NMS)$  and  $nTclk_{\min}(NMS)$  of the proposed organization of computations cannot be tested on a comparative basis

- The specifics of this Canny module requires two two levels of presenting test results for  $nTclk_{\min}$ :

- A)  $nTclk_{\min}(NMS)$  which defines the minimum number of clock cycles required to process a single image pixel with non-maximum suppression;

B)  $nTclk_{min}(Thresh)$  which defines the minimum number of clock cycles required to compute high and low thresholds in a dynamic fashion.

The achieved results are shown in Table 3.

<b>Table 3. <math>F_{max}(NMS)</math>, <math>nTclk_{min}(NMS)</math> and <math>nTclk_{min}(Thresh)</math> of the designed organization of non-maximum suppression with dynamic computation of high and low thresholds</b>			
<b>FPGA family</b>	<b>Speed of the designed organization of non-maximum suppression with dynamic computation of high and low thresholds</b>		
	<b><math>F_{max}(NMS)</math> (in MHz)</b>	<b><math>nTclk_{min}(NMS)</math> required to non-maximum suppress a single image pixel</b>	<b><math>nTclk_{min}(Thresh)</math> required to calculate the low threshold</b>
<b>Cyclone</b>	232	1	39
<b>Cyclone II</b>	319	1	39
<b>Cyclone III</b>	367	1	39
<b>Cyclone iV</b>	370	1	39
<b>Cyclone V</b>	388	1	39
<b>Stratix</b>	331	1	39
<b>Stratix II</b>	464	1	39
<b>Stratix III</b>	589	1	39
<b>Stratix IV</b>	637	1	39
<b>Stratix V</b>	685	1	39

#### 4. Analysis of results

The obtained experimental data prove that:

- Because the largest value in a counter is guaranteed to be within 17 bits on the basis of (8) and (4)), the data in Table 3 represents the worst case scenario for  $F_{max}(NMS)$ . Therefore, according to the data in [4] [5] [6] [7] [8] is guaranteed that (1) is satisfied under all test conditions.
- Because the comparisons of the central pixel in the square neighbourhood with the two adjacent pixels along gradient direction are executed in parallel, it is guaranteed that (2) is satisfied under all test conditions. Therefore, the requirements of mid-level pipelining, low level parallelism and the number of uses of a single pixel expressions are satisfied.
- The value of  $nTclk(pauseTotalNMS)$  in (4) is a quantitative measurement of the decrease in mid-level pipelining efficiency. It is the sum of empty clock cycles for all image pixel except pixel #1.
- $nTclk_{min}(Thresh)$  requires only the last 39 clock cycles of the execution of non-maximum suppression module of FPGA based gradient edge detection. Therefore, the non-maximum suppression computations and the threshold computations end simultaneously. Thus, the requirements are satisfied.

#### 5. Conclusions

Proposed in this paper is a non-maximum suppression technology with simultaneous dynamic threshold computation. The upper speed limit of the proposed organization of computations is

experimentally investigated in ten Intel (Altera) FPPA families. On the basis of applying mid-level pipelining, low level parallelism and the number of uses of a single pixel approaches for minimizing the required number of clock cycles it is ascertained that the entire threshold computation takes only 39 clock cycles to execute. Most importantly, the non-maximum suppression computations and the threshold computation end simultaneously under all test conditions.

## References

- [1]. Canny J., A Computational Approach to Edge Detection, IEEE Transaction on Pattern Analysis and Machine intelligence, No. 6, 1986, pp. 679-698
- [2]. Cao, Yi qin, Wu Dan, Dan Duan, Yeyu, A new image edge detection algorithm based on improved Canny, Journal of Computational Methods in Sciences and Engineering, vol. 20, no. 2, 2020, pp. 629-642
- [3]. Dimitre Kromichev, Gradient Based Edge Detection: Investigating the Problem of False Negatives in Terms of Edge Thinning, Proceedings of 16th International Engineering Conference on Communications, Electromagnetics and Medical Applications (CEMA'23), 2023, pp. 52-57
- [4]. D. Kromichev, FPGA Based Canny: Advanced Integer Division Algorithm, 2021 12th National Conference with International Participation (ELECTRONICA), Sofia, Bulgaria, 2021, pp. 1-4
- [5]. Dimitre Kromichev, FPGA Based Edge Detection: Integer Square Root Algorithm, Proceedings of 16th International Engineering Conference on Communications, Electromagnetics and Medical Applications (CEMA'22), 2022, pp. 25-29
- [6]. Dimitre Kromichev An Approach to Computing the Gradients in FPGA Based Edge Detection Focused on Ultimate Execution Speed and Exactness of Detected Contours, Proceedings of 16th International Engineering Conference on Communications, Electromagnetics and Medical Applications (CEMA'23), 2023, pp. 52-57
- [7]. D. Kromichev, FPGA Based Edge Detection: Integer Division Algorithm with a Constant Divisor, 2022 13th National Conference with International Participation (ELECTRONICA), Sofia, Bulgaria, 2022, pp. 1-4
- [8]. Dimitre Kromichev, The Experimental Basis of Investigations in FPGA Based Gradient Edge Detection with Respect to Accurate Calculation of Gaussian and Sobel filters, Computer Science and Technologies, ISSN 1312- 3335, Vol. 1, 2023, pp. 58-67
- [9]. Gao Jie and Liu Ning, An improved adaptive threshold canny edge detection algorithm, IEEE Transactions, International Conference on Computer Science and Electronics Engineering (ICCSEE), Vol. 1, 2012, pp.164-168
- [10]. Kewen Liu, Kang Xiao, and Hongxia Xiong, An Image Edge Detection Algorithm Based on Improved Canny, 5th International Conference on Machinery, Materials and Computing Technology (ICMMCT 2017), 2017, pp. 533-537
- [11]. Lin Zhang, Image Adaptive Edge Detection Based on Canny Operator and Multiwavelet Denoising, 2014 International Conference on Computer Science and Electronic Technology (ICCSET 2014), 2014, pp. 335-338
- [12]. L. Rao, Bin Zhang, Jizhong Zhao, Hardware Implementation of Reconfigurable 1D Convolution, Journal of Signal Processing Systems, 2016, pp. 232-237
- [13]. Ojashwini R.N., Gangadhar Reddy R., Rani R.N., Pruthvija B., Edge Detection Canny Algorithm Using Adaptive Threshold Technique, Intelligent Data Engineering and Analytics. Advances in Intelligent Systems and Computing, vol. 1177, 2021, pp. 202-210

- [14]. Rinjo A. J. and Remya K. P., FPGA Implementation of Distributed Canny Edge Detector for Low Clarity Images, IJSET - International Journal of Innovative Science, Engineering & Technology, Vol. 3, Issue 6, June 2016, pp. 618-622
- [15]. R. Jayarani, Edge Detection using Distributed CANNY Algorithm and Implementation in FPGA, International Journal of Research in Engineering, Science and Management, Volume-2, Issue-5, May-2019, pp. 1043-1047
- [16]. Shwetha.V., C.H.Renu Madhavi, Design Techniques for Improvement of Canny Edge Detection Algorithm, International Journal of Creative Research Thoughts (IJCRT), Volume [8, Issue 8, August 2020, pp. 1224-1231
- [17]. SunTao and Chang Zhi Gao, An Improved Canny Edge Detection Algorithm, Applied Mechanics and Materials, vol. 291–294, Trans Tech Publications, Ltd., Feb. 2013, pp. 2869–2873
- [18]. Tao Yang, Yue-hong Qiu, Improvement and implementation for Canny edge detection algorithm, Proc. SPIE 9631, Seventh International Conference on Digital Image Processing (ICDIP 2015), 6 July 2015, pp. 845-854
- [19]. Xiang Wu, Weibo Yu, Xiaotong Liu and Keping Liu, A Newly Improved Canny Algorithm of Image Edge Detection, International Conference on Information Engineering for Mechanics and Materials (ICIMM 2016), 2016, pp. 501-510

Dimitre Zhivkov Kromichev, PhD  
Department of Marketing and International Economic Relations  
University of Plovdiv  
e-mail: [dkromichev@yahoo.com](mailto:dkromichev@yahoo.com)





# ULTIMATE EXECUTION SPEED IN FPGA BASED GRADIENT EDGE DETECTION: TECHNOLOGY OF ACCURATE COMPUTATION OF GRADIENT DIRECTION

Dimitre Kromichev

**Abstract:** FPGA based edge detection targeting ultimate execution speed and relying on gradient direction to define image contours has to tackle the problem of computing gradient direction accurately and fast. This paper presents an organization of computations in gradient direction designed to be used in the gradient direction submodule of FPGA based gradient edge detection. Having investigated the mathematical accuracy, maximum operating frequency and minimum number of clock cycles on the basis of ten Intel (Altera) FPGA families, it is ascertained that the proposed organization of gradient direction computations is capable of securing total accuracy and working at a frequency higher than the maximum operating frequency of embedded memory under all test conditions. It takes only three clock cycles to provide an accurate result.

**Keywords:** FPGA, edge detection, gradient direction, inverse tangent, operating frequency, clock cycle

## 1. Introduction

In [3] an FPGA based design targets the speeding up of  $\tan^{-1}$  computations. The results are: execution time is 320 ns; accuracy is  $<0.01^\circ$ . Taylor series expansion is applied to transfer  $\tan^{-1}$  to a polynomial form [3][11]. In [6] two-argument  $\tan^{-1}$  is implemented in FPGA by using the piecewise polynomial approximation with non-uniform segmentation. The inputs (x and y) are divided using radix-2 non-restoring division and the result is used as an input to  $\text{Atan}$ . The results are: maximum error ratio - 2.62%; execution time in Spartan 6 - 260.5 ns. In FPGA based design of two-argument  $\tan^{-1}$  [9][8][10] division of the two inputs is implemented by a logarithmic transformation using subtraction. In [2] studied is the FPGA implementation of fixed-point two-argument  $\tan^{-1}$  by comparing CORDIC with two multiplier based techniques. In [12] presented are several approximations for four quadrant  $\tan^{-1}$  using Lagrange interpolation and optimization techniques. In [5] it is pointed out that long latency is a main disadvantage of methods based on CORDIC, conventional LUTs and polynomial approximation. In [1] proposed is  $\text{atan2}$  using look-up table with 101-points. The accuracy is increased by linear interpolation. The achieved frequency is 60 MHz. In [4] proposed is a high-accuracy computation of fixed-point  $\tan^{-1}$  using CORDIC and fast magnitude estimation. Maximum phase error is reduced from 414 LSB (angle error of 0.6355 rad) to 4 LSB (angle error of 0.0061 rad). In [7] described is an FPGA implementation of  $\tan^{-1}$  which is based on using serial and pipelined CORDIC architectures.

The objective of this paper is to propose an organization of gradient direction computation in FPGA based edge detection. The mathematical accuracy, maximum operating frequency and minimum number of clock cycles of gradient direction are investigated using ten Intel (Altera) FPGA families. The employed tools are: Scilab, Intel (Altera) Quartus, VHDL, TimeQuest Timing Analyzer, ModelSim. The analyses and conclusions are relevant to gray scale images.

## 2. Integer inverse tangent computations

Because

$$\begin{aligned}
y &= \tan x \Leftrightarrow \tan^{-1} y = x & (1) \\
x &\in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right) \\
y &\in R
\end{aligned}$$

in the integer domain of FPGA based gradient edge detection the expression

$$\{(y, x) | y = \tan x, x \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right), y \in R\} \quad (2)$$

represents a problem which must be solved according to the specifics of gradient direction.

The goal of inverse tangent is to guarantee that two conditions are satisfied:

- $F_{\max}(\tan^{-1} New) > F_{\max}(mem) \quad (3)$

where

$F_{\max}(\tan^{-1} New)$  is the maximum operating frequency of inverse tangent

- $nTclk(\tan^{-1} New) \leq 4 \quad (4)$

where

$nTclk(\tan^{-1} New)$  is the number of clock cycles required by inverse tangent to calculate.

Because in a digital image matrix every pixel has 8 adjacent pixels, there are 4 pairs of opposite pixels, and each pair defines its own axis (Figure 1). The obtained four axes represent bisectors of 4 angular sectors, with each sector encompassing 45°. Therefore, all gradient direction values are: 0, 90, 45, 135. Hence, the equations:

$$\tan^{-1} \left( \frac{G_y}{G_x} \right) = 0 \quad (5)$$

$$\tan^{-1} \left( \frac{G_y}{G_x} \right) = 45 \quad (6)$$

$$\tan^{-1} \left( \frac{G_y}{G_x} \right) = 90 \quad (7)$$

$$\tan^{-1} \left( \frac{G_y}{G_x} \right) = 135 \quad (8)$$

Solving (5), (6), (7) and (8) requires finding the domains of four functions with predefined fixed ranges:

$$A_0 = \tan^{-1} \left( \frac{m_0}{n_0} \right) \quad (9)$$

$$A_{45} = \tan^{-1} \left( \frac{m_{45}}{n_{45}} \right) \quad (10)$$

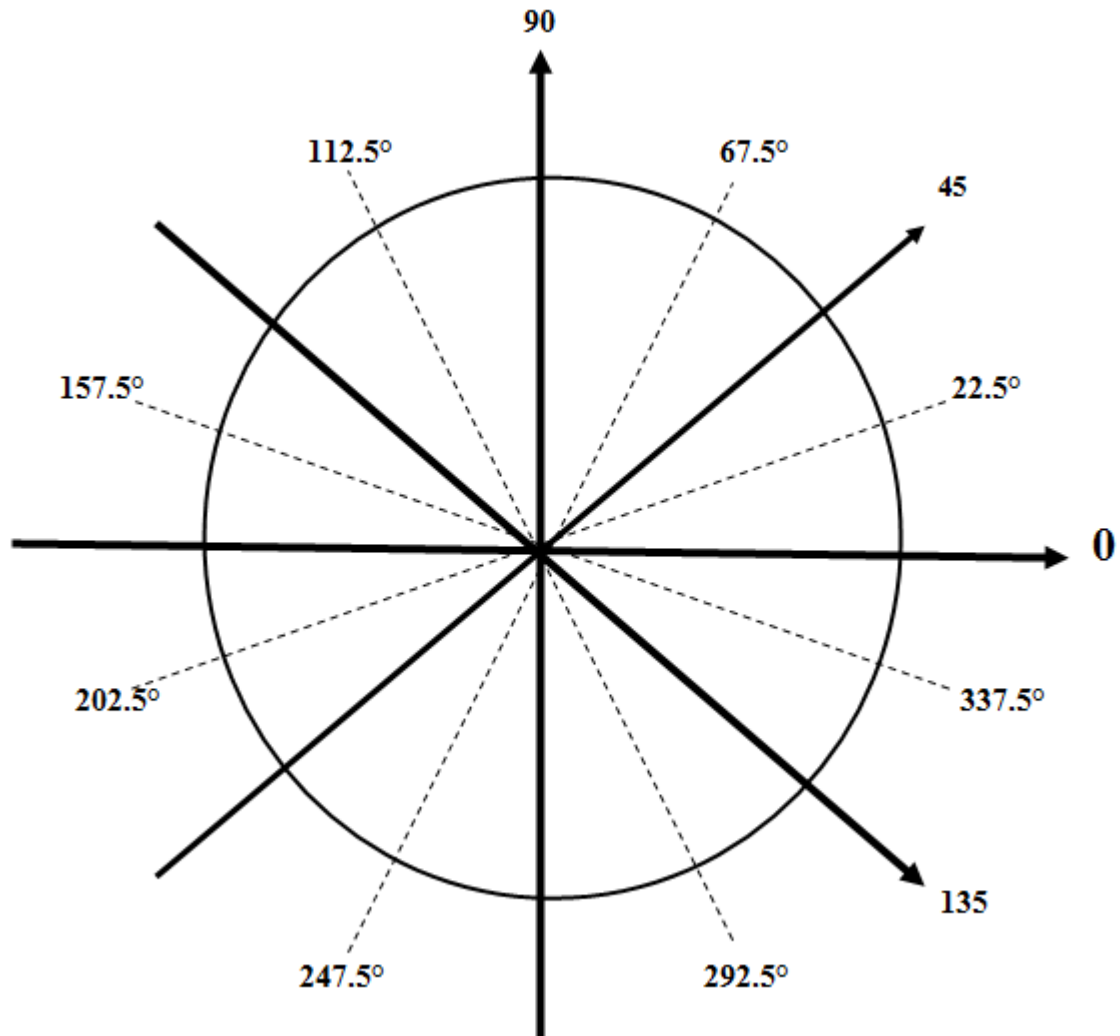
$$A_{90} = \tan^{-1} \left( \frac{m_{90}}{n_{90}} \right) \quad (11)$$

$$A_{135} = \tan^{-1} \left( \frac{m_{135}}{n_{135}} \right), \quad (12)$$

where

$A_0$  is the angular sector containing the axis for 0,

$A_{45}$  is the angular sector containing the axis for 45,



**Fig. 1.** Unit circle for gradient direction

$A_{90}$  is the angular sector containing the axis for 90,  
 $A_{135}$  is the angular sector containing the axis for 135,  
 $m_0, n_p$  are independent variables,  $m_0 \in [-255, 255]$ ,  $n_p \in [-255, 255]$   
 $m_{45}, n_{45}$  are independent variables,  $m_{45} \in [-255, 255]$ ,  $n_{45} \in [-255, 255]$ ,  
 $m_{90}, n_{89}$  are independent variables,  $m_{90} \in [-255, 255]$ ,  $n_{89} \in [-255, 255]$ ,  
 $m_{135}, n_{135}$  are independent variables,  $m_{135} \in [-255, 255]$ ,  $n_{135} \in [-255, 255]$ .

Therefore, the task is to define the complete set of values of the independent variables in (9), (10), (11) and (12). In order to avoid division by 0 division must not be used

- Due to total symmetry the axes for 0, 45, 90 and 135 are further divided into two pairs. The ingredients of each pair are orthogonal. Hence, the angular sector for 45 and the angular sector for 135 are symmetric with respect to the x-axis. Therefore

$$\begin{aligned}
 |m_{45}| &= |m_{135}| \\
 |n_{45}| &= |n_{135}|.
 \end{aligned} \tag{13}$$

As a result, the difference between 45 and 135 is based on sign relations.

- Unlike 45 and 135, the difference between 0 and 90 is defined by the fact that their angular sectors are symmetrical with respect to axis 45. Hence, the sets of  $|m_0|, |n_0|, |m_{45}|, |n_{45}|, |m_{90}|, |n_{90}|, |m_{135}|, |n_{135}|$  are determined on the basis of comparison with respect to the boundaries of the angular sector for 45.

### 3. Computational mechanism of inverse tangent in FPGA

Integer inverse tangent implements parallelism. Its computational mechanism includes:

Step #1 Determine all possible combinations between the signs of  $G_y$  and  $G_x$

Step #2 Divide combinations from *Step #1* into two sets:

$$\begin{aligned} G_y > 0 \ \& \ G_x > 0 \\ G_y < 0 \ \& \ G_x < 0 \end{aligned} \quad (14)$$

and

$$\begin{aligned} G_y > 0 \ \& \ G_x < 0 \\ G_y < 0 \ \& \ G_x > 0. \end{aligned} \quad (15)$$

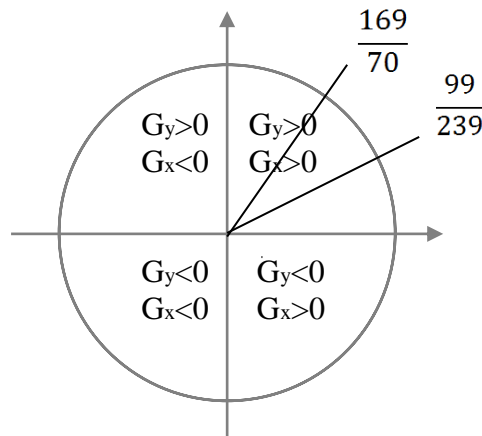
Step #3 Define two reference points:  $22.5^\circ$  and  $67.5^\circ$ .

Step #4 Determine a numerical equivalent to angle  $22.5^\circ$ . The accurate representation of angle

$$22.5^\circ \text{ is the fraction } \frac{99}{239} \left( \tan^{-1} \left( \frac{99}{239} \right) = 22.500605394851^\circ \right)$$

Step #5 Determine a numerical equivalent to angle  $67.5^\circ$ . The accurate representation of angle

$$67.5^\circ \text{ is the fraction } \frac{169}{70} \left( \tan^{-1} \left( \frac{169}{70} \right) = 67.500605394851^\circ \right) \text{ (Figure 2).}$$



**Fig. 2.** The two reference points and sign options for inverse tangent

Step #6 According to the sign relations defined in (14) calculations for obtaining the value of gradient direction  $Dir$  are executed simultaneously according to

$$\begin{aligned} &G_y > 0 \ \& \ G_x > 0 \\ &G_y < 0 \ \& \ G_x < 0 \\ \text{If } &|G_x| * 99 \geq |G_y| * 239 & \quad Dir = 0 \\ \text{If } &|G_x| * 99 < |G_y| * 239 \ \& \ |G_x| * 169 > |G_y| * 70 & \quad Dir = 45 \\ \text{If } &|G_x| * 169 \leq |G_y| * 70 & \quad Dir = 90 \end{aligned} \quad (16)$$

Step #7 According to the sign relations defined in (15) calculations for obtaining the value of gradient direction  $Dir$  are executed simultaneously according to

$$\begin{aligned}
 &G_y > 0 \ \& \ G_x < 0 \\
 &G_y < 0 \ \& \ G_x > 0 \\
 \text{If } &|G_x| * 99 \geq |G_y| * 239 & \quad Dir = 0 \\
 \text{If } &|G_x| * 99 < |G_y| * 239 \ \& \ |G_x| * 169 > |G_y| * 70 & \quad Dir = 135 \\
 \text{If } &|G_x| * 169 \leq |G_y| * 70 & \quad Dir = 90 .
 \end{aligned} \tag{17}$$

Step #8 In case of orthogonal gradients equal 0

$$\begin{aligned}
 \text{If } &G_y = 0 \ \& \ G_x \neq 0 & \quad Dir = 0 \\
 \text{If } &G_y \neq 0 \ \& \ G_x = 0 & \quad Dir = 0 \\
 \text{If } &G_y = 0 \ \& \ G_x = 0 & \quad Dir = 0 .
 \end{aligned} \tag{18}$$

The computational mechanism in FPGA is presented in Figure 3. Mathematical accuracy is tested for all values of  $G_y$  and  $G_x$  in the interval  $[-255, 255]$  and sign relations using Scilab. Four sample test results are presented below. Accuracy is best proved by using ratios of  $G_y$  and  $G_x$  values which are close to the two reference points -  $\frac{99}{239}$  and  $\frac{169}{70}$ , representing the two critical angles -  $22.5^\circ$  and  $67.5^\circ$ .

#### Check # 1

$$\begin{aligned}
 &G_y = -160 \\
 &G_x = -66 \\
 &|-66| * 99 \geq |-160| * 239 & \quad (\text{false}) \\
 &|-66| * 99 < |-160| * 239 \ \& \ |-66| * 169 > |-160| * 70 & \quad (\text{false}) \\
 &|-66| * 169 \leq |-160| * 70 & \quad (\text{true}) \\
 \text{Therefore, } &Dir = 90 .
 \end{aligned}$$

Using the conventional method:  $\tan^{-1} \left( \frac{-160}{-66} \right) = 67.583852520656^\circ$ .

#### Check # 2

$$\begin{aligned}
 &G_y = -48 \\
 &G_x = 55 \\
 &55 * 99 \geq |-48| * 239 & \quad (\text{false}) \\
 &55 * 99 < |-48| * 239 \ \& \ 55 * 169 > |-48| * 70 & \quad (\text{true}) \\
 &55 * 169 \leq |-48| * 70 & \quad (\text{false}). \\
 \text{Therefore, } &Dir = 135 .
 \end{aligned}$$

Using the conventional method:  $\tan^{-1} \left( \frac{-48}{55} \right) = -41.1120904^\circ$ .

#### Check # 3

$$\begin{aligned}
 &G_y = -19 \\
 &G_x = -46 \\
 &|-46| * 99 \geq |-19| * 239 & \quad (\text{true}) \\
 &|-46| * 99 < |-19| * 239 \ \& \ |-46| * 169 > |-19| * 70 & \quad (\text{false})
 \end{aligned}$$

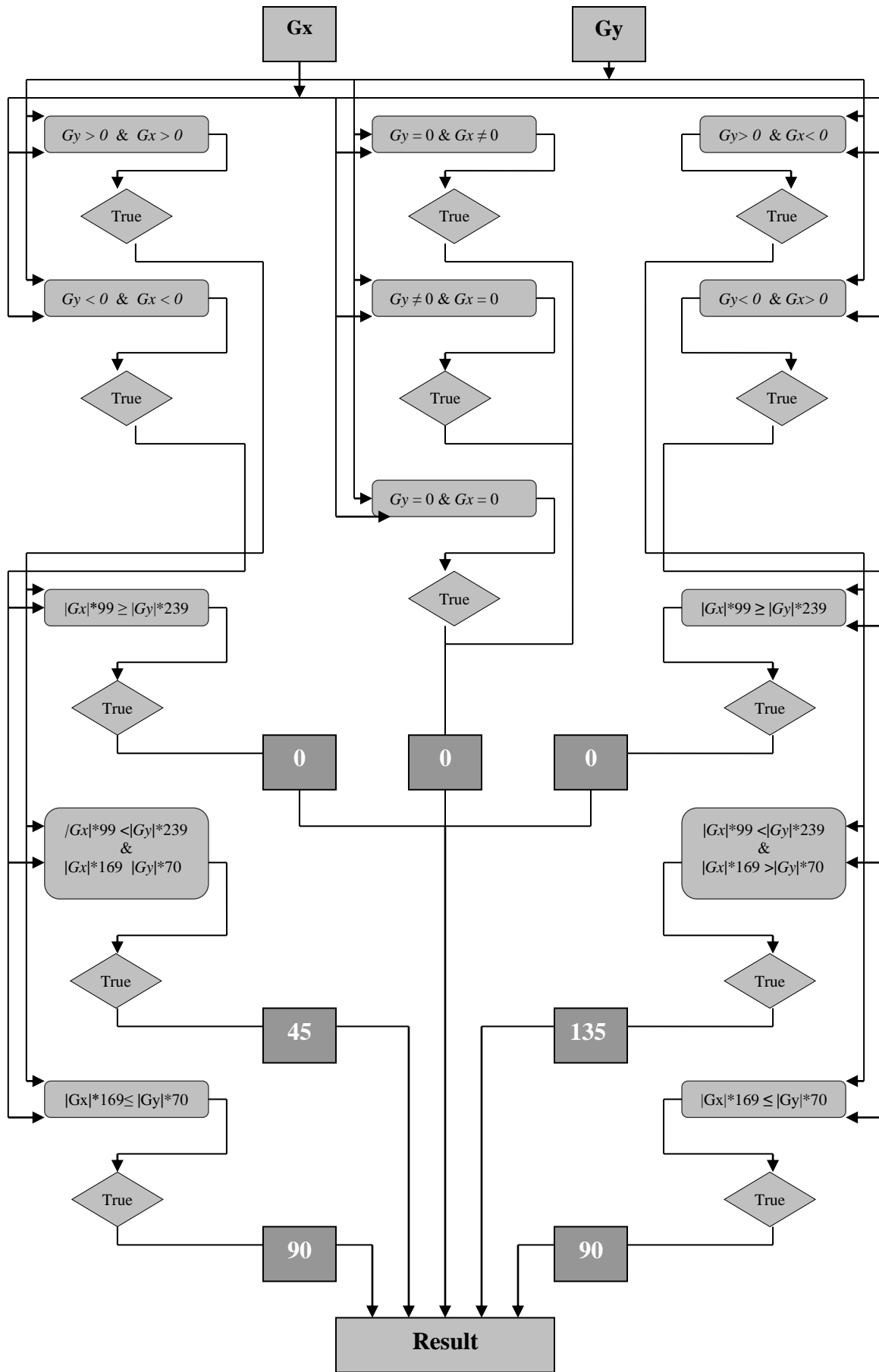


Fig. 3. Model of computational mechanism of integer inverse tangent

$$|-46|*169 \leq |-19|*70 \quad (\text{false})$$

Therefore,  $Dir = 0$ .

Using the conventional method:  $\tan^{-1}\left(\frac{-19}{-46}\right) = 22.442753365294^\circ$ .

#### Check # 4

$$Gy = 19$$

$$Gx = 45$$

$$|45|*99 \geq |19|*239 \quad (\text{false})$$

$$|45|*99 < |19|*239 \ \& \ |45|*169 > |19|*70 \quad (\text{true})$$

$$|45|*169 \leq |19|*70 \quad (\text{false})$$

Therefore,  $Dir = 45$ .

Using the conventional method:  $\tan^{-1}\left(\frac{19}{45}\right) = 22.890551656248^\circ$ .

These results prove that the two reference points  $-\frac{99}{239}$  and  $\frac{169}{70}$ , are accurately calculated, and the algorithm guarantees total mathematical accuracy.

#### **4. FPGA based gradient direction: functionality and RTL design**

Gradient direction computations are based on inverse tangent. Therefore, the organization of computations focused on optimal speed satisfies the two basic requirements:

- Maximum operating frequency of gradient direction  $F_{\max}(D)$
- Minimum number of clock cycles  $nTclk_{\min}(D)$ .

The functional model of FPGA based gradient direction is in Figure 4. The RTL design of gradient direction is presented in Figure 5. Resource utilization of gradient direction is presented in Table 1.

#### **5. Exploring the speed of gradient direction computations. Analysis of results**

The achieved speed results are shown in Table 2. Test results prove the total mathematical accuracy of calculated values. The input data widths  $\leq 8$  bits for both the numerator and denominator in the two reference points  $\frac{99}{239}$  and  $\frac{169}{70}$ . The largest input data width for a gray scale image pixel is 8 bits. Therefore, for Cyclone II-V and Stratix I-V,  $F_{\max}$  of integer inverse tangent is defined by  $F_{\max}$  of 9x9 hard multiplier. For Cyclone,  $F_{\max}$  of integer inverse tangent is defined by  $F_{\max}$  of 8x8 logic elements based multiplier.

Therefore, on the basis of  $F_{\max}$  and  $nTclk_{\min}$ , integer inverse tangent is optimal for all input data widths calculated in FPGA based edge detection.

Data in Table 2 show that in Cyclone II-V and Stratix I-V  $F_{\max}(D)$  is defined only by  $F_{\max}(\text{hard})$  of 9x9 multiplier. In Cyclone,  $F_{\max}(D)$  is defined only by  $F_{\max}$  of 8x8 logic elements based multiplier.  $nTclk_{\min}(D) = 3$ .

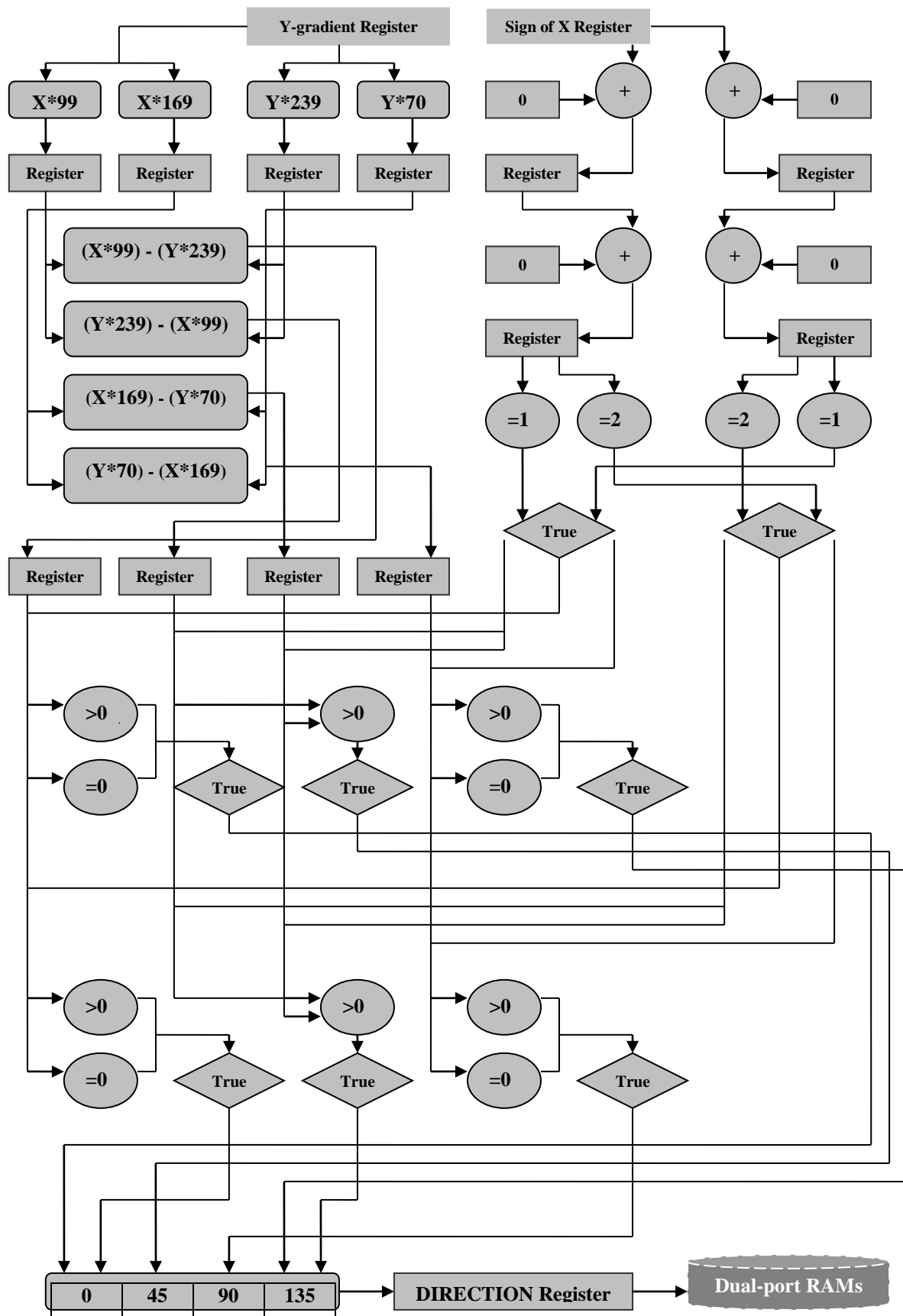


Fig. 4. Functional model of gradient direction



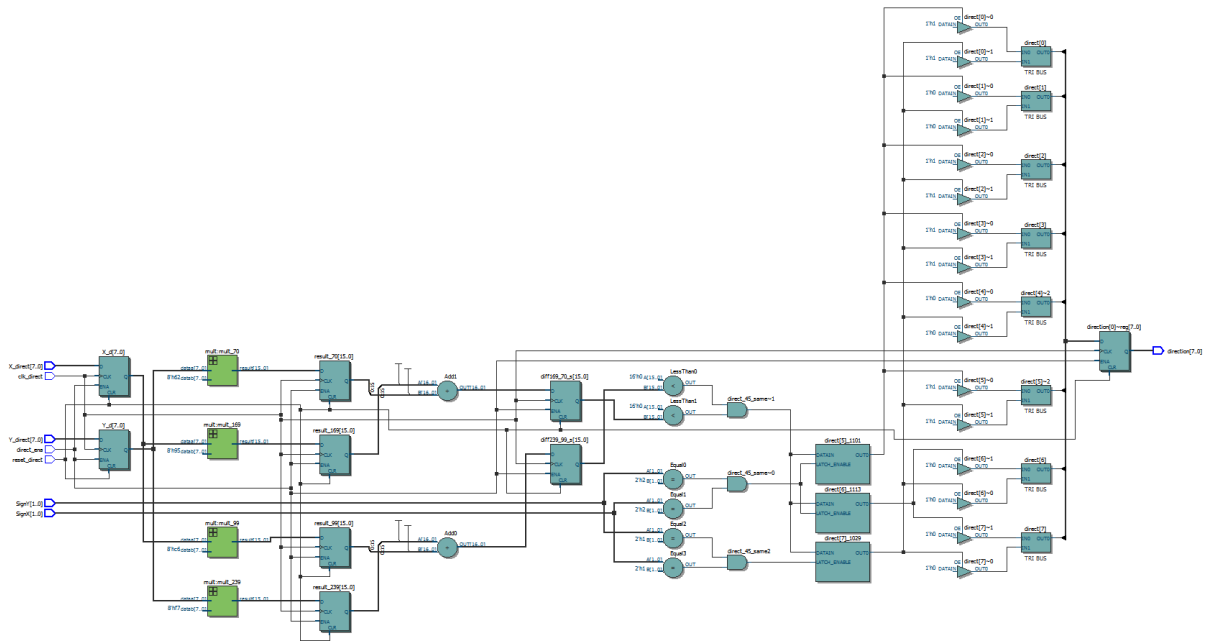


Fig. 5. RTL design of gradient direction (Source: Intel (Altera) Quartus RTL Viewer)

Table 1. Resource utilization of gradient direction					
FPGA family	Logic utilization	Total registers	Total memory bits	Total DSP blocks	Embedded mult 9 bit
Cyclone	122 (LEs)	-	0	-	-
Cyclone II	104 (LEs)	82	0	-	4
Cyclone III	93 (LEs)	50	0	-	4
Cyclone IV	93 (LEs)	50	0	-	4
Cyclone V	36 (ALMs)	50	0	4	-
Stratix	116 (LEs)	84	0	3	-
Stratix II	82 (ALUTs)	70	0	3	-
Stratix III	69 (ALUTs)	48	0	3	-
Stratix IV	69 (ALUTs)	48	0	3	-
Stratix V	34 (ALMs)	48	0	2	-

Table 2. $F_{\max}(D)$ and $nTclk_{\min}(D)$ of gradient direction		
FPGA family	$F_{\max}(D)$ (in MHz)	$nTclk_{\min}(D)$
Cyclone	179	3
Cyclone II	259	3
Cyclone III	331	3
Cyclone iV	334	3
Cyclone V	337	3
Stratix	313	3
Stratix II	428	3
Stratix III	534	3
Stratix IV	538	3
Stratix V	540	3

## 5. Conclusions

This paper presents integer inverse tangent and gradient direction computations. Their application is FPGA based edge detection which targets ultimate execution speed. The designed tangent and gradient direction computations are explored for mathematical accuracy, maximum operating frequency and minimum number of clock cycles in ten Intel (Altera) FPGA families. The maximum operating frequency in Cyclone II-V and Stratix I-V is defined only by maximum operating frequency of 9x9 multiplier. In Cyclone, maximum operating frequency is defined only by the maximum operating frequency of 8x8 logic elements based multiplier. The minimum number of clock cycles is 3.

## References

- [1]. Ukil, V. H. Shah, and B. Deck, Fast computation of arctangent functions for embedded applications: A comparative analysis, 2011 IEEE International Symposium on Industrial Electronics, 2011, pp. 1206-1211
- [2]. F. De Dinechin and M. Istoan, Hardware Implementations of Fixed-Point Atan2, 2015 IEEE 22nd Symposium on Computer Arithmetic, 2015, pp. 34-41
- [3]. Kung, Y.-S., Wu, M.-K., Linh Bui Thi and, H., Jung, T.-H., Lee, F.-C., and Chen, W.-C., FPGA-based hardware implementation of arctangent and arccosine functions for the inverse kinematics of robot manipulator, Engineering Computations, Vol. 31 No. 8, 2014, pp. 1679-
- [4]. Luca Pilato, Luca Fanucci, and Sergio Saponara, Real-Time and High-Accuracy Arctangent Computation Using CORDIC and Fast Magnitude Estimation, Electronics, 2017, pp. 6-22
- [5]. M. Saber, Y. Jitsumatsu, and T. Kohda, A low-power implementation of arctangent function for communication applications using FPGA, 2009 Fourth International Workshop on Signal Design and its Applications in Communications, 2009, pp. 60-63
- [6]. Omar Zeyad, Design and Implementation Hardware Architecture for Four-Quadratic Arctangent, International Journal of Computer Applications, Volume 176 – No.1, October 2017, pp. 10-13
- [7]. P. A. Kumar, FPGA Implementation of the Trigonometric Functions Using the CORDIC Algorithm, 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), 2019, pp. 894-900
- [8]. Roberto Gutierrez and Javier Valls, Low-power FPGA-implementation of atan (y/x) using look-up table methods for communication applications, Journal of Signal Processing Systems, Volume 56, Issue 1, July 2009, pp. 25–33
- [9]. R. Gutierrez and J. Valls, Implementation on FPGA of a LUT-Based atan(Y/X) Operator Suitable for Synchronization Algorithms, 2007 International Conference on Field Programmable Logic and Applications, 2007, pp. 472-475
- [10]. R. Gutierrez, V. Torres, and J. Valls, FPGA-implementation of atan(Y/X) based on logarithmic transformation and LUT-based techniques, Journal of Systems Architecture, vol. 56, 2010
- [11]. S. Nandi, S. Prasad, C. M. Ananda, and S. S. Rekha, Fixed point implementation of trigonometric function using Taylor's series and error characterization, 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2016, pp. 442-446
- [12]. S. Rajan, S. Wang, and R. Inkol, Efficient Approximations for the Four-Quadrant Arctangent Function, 2006 Canadian Conference on Electrical and Computer Engineering. IEEE, 2006

Dimitre Zhivkov Kromichev, PhD  
Department of Marketing and International Economic Relations  
University of Plovdiv  
e-mail: [dkromichev@yahoo.com](mailto:dkromichev@yahoo.com)

# МОДЕЛИРАНЕ НА ЙЕРАРХИЧНА СТРУКТУРА ЗА НОРМАТИВНА УРЕДБА ПО ИНФОРМАЦИОННО И КОМУНИКАЦИОННО ОСИГУРЯВАНЕ

Георги Николов

**Резюме:** Статията обръща внимание върху подхода по приоритетното постигане на цели, засягащи необходимостта от познаването, бързото ориентиране и точното спазване на минимума от публично достъпната нормативна уредба (определена, конкретизирана и актуализирана) на международното (междудържавно), националното (държавно) и секторно-ориентирано (специализирано) равнища. В условия на ограничено време и необходимост от съгласуване и справки при използване на множество регламентиращи документи в юридически и технически план, следва и се акцентира на основните хармонизирани причинно-следствени връзки, без да се навлиза в посока на излишно детайлизиране и разсъждения по извънтематичните им особености. Във връзка с това, обект на настоящите изследвания е съобразяване на възможностите и оптимален избор на съвременни инструменти и информационни структури (системи), отговарящи на съответните критерии и подпомагащи функциите на ресорни органи и длъжностни лица.

**Ключови думи:** модел, моделиране, информация, осигуряване, граф, множество, релация, йерархична структура, нормативна уредба, законодателство, стандарти, база данни, SQL, таблица, съдържание, заявка.

## Modelling of a hierarchical structure for legal regulation on information and communication provision

Georgi Nikolov

**Abstract:** The article draws attention to the approach of priority achievement of goals affecting the need for knowledge, quick orientation and exact compliance with the minimum of the publicly available regulatory framework (defined, specified and updated) of the international (interstate), national (state) and sector-oriented (specialized) levels. In conditions of limited time and the need for coordination and references when using multiple regulatory documents in legal and technical terms, the main harmonized cause-and-effect relationships are followed and emphasized, without going into the direction of unnecessary detailing and reasoning on their off-topic features. In this regard, the object of the current research is consideration of the possibilities and optimal selection of modern tools and information structures (systems), meeting the relevant criteria and supporting the functions of departmental persons and officials.

**Keywords:** Model, Modeling, Information, Provision, Graph, Set, Relation, Hierarchical structure, Regulatory, Legislation, Standards, Database, SQL, Table, Content, Query.

### 1. Въведение

В математическото моделиране се налагат акцентите по изучаването на класове от математически структури като групи, полета, графите и теорията на множествата, използвайки инструментите на математическата логика. Ефективната теория на моделите, наричана още ефективна структурна теория, изучава взаимодействието между структурите и тяхната сложност. Под структури се има предвид обекти като групи, пръстени, полета, графи или линейни наредби, които имат непразен носител, върху който са дефинирани различни релации, функции и константи [1]. Теорията на моделите е близка до универсалната алгебра и теорията на вероятностите. Графът се разглежда като съвкупност от върхове (възли) и дъги (ребра). Върховете (със степен 1) се наричат висящи върхове на графа, а върховете (със

степен 0) се наричат изолирани върхове [2]. Множество се нарича съвкупност от определени различни един от друг елементи [3]. Даден вид обект може да бъде поместен в произволно множество, но теорията на множествата, прилагана обичайно за обекти, свързани с математиката, позволява безпристрастни заключения и изводи. Термините от теорията на множествата може да се използват в определенията на почти всички математически обекти. След откриването на ограничена изчерпателност са предложени вариативни аксиоматични системи, от които най-познатите са системата на Цермело-Френкел, съответствие (Декартови произведения, Релации, Функции) и аксиомата за избора [4]. Елементарни факти за множества и принадлежност към множества често се използват заедно с различни инструменти. За графично представяне на булеви операции с множества е удобно да се използват диаграми на Вен [3]. С помощта на метод на Ойлер-Вен диаграми се визуализират множествата и съответните класове, докато с метод на MZ-картите се визуализират числата и сумиращите операции.

Актуалните изследвания, свързани с теорията на множествата, включват разнообразна съвкупност от теми, обхващащи задачи от структурата на реалните числови редици до проучване и развиване на сложни и обемни такива. Основни релации между две множества са релацията „подмножество“, релацията „равенство“ и релацията „строго подмножество“ [3]. Както при бинарните операции върху числа в аритметиката, теорията на множествата използва бинарни операции върху множества:

- сечение на множества;
- обединение на множества;
- разлика и допълнение на множества;
- обобщено обединение и сечение на множества.

Основни множества с голямо значение в математиката са празното множество, което не съдържа никакви елементи и е подмножество на всяко множество, както и множествата на естествените, целите, рационалните, реалните и комплексните числа [3]. Термините йерархична и линейна структура се отнасят до начина, по който една логическа, административна или техническа единица осигурява такава във възходящ ред и как контролира своите низходящи такива, включително в обхват на предоставени права и отговорности. Всяка една от двете структури има своите предимства и недостатъци. В най-общо определение йерархия е разположение на членовете (елементите, единиците) на сложна система, подредени последователно по ранг (или по степен на сложност) от най-висшите към най-нисшите [5]. Понятието йерархия е с приложение във всички области на научен интерес. Сред тях са теорията за езиковите равнища и граматическите теории като фразово-структурната граматика [5]. Предимството на йерархичния тип система е качественият контрол върху работните цикли с действащите алгоритми и структури.

Линейната система на управление не разполага с много нива на отговорности, което прави и броят на началници и мениджъри ограничен. Много често между служителят във фирмата и директорът може да има само едно ниво, което да ги дели. Линейната и йерархичната структури са условни понятия [6]. Анализът им трябва да бъде съпроводен от сравняване със система, съизмерима по обхват и сериализация на функции. Системите с йерархична структура на управление са чувствителни по отношение на стъпаловидното акумулиране на време и изчислителни ресурси. При осъществяването на техническата стандартизация и осигуряването на еднозначни изисквания по задания, проектиране, разработване на изделия, системи и среди се прилагат нормативни актове в международен, национален, партньорски и вътрешен (корпоративен) план. Дейностите, налагащи прилагането, могат да обхванат:

- оперативно планиране и синхронизиране на нормативната база;
- поддържане в актуално състояние, съхраняване и архив на документацията;
- защита на конфиденциалност и авторски права;

- управление на процеси, технологии и оборудване;
- мониторинг и контрол за съответствието и чувствителността на информацията;
- технологично осигуряване, модернизация, ремонт и възстановяване;
- разработване на механизми за адаптивност в публична и корпоративна среда;
- оценка и цялостен анализ на финансовата и енергийна ефективност в глобален и регионален мащаб.

За постигането на изброените дейности неминуемо се налага дълбоко осмисляне и обосноваване прилагане на най-подходящите инструменти, които гарантират трайни и общопризнати резултати.

## **2. Структуриране на модел за правно-нормативната уредба по комуникационно и информационно осигуряване**

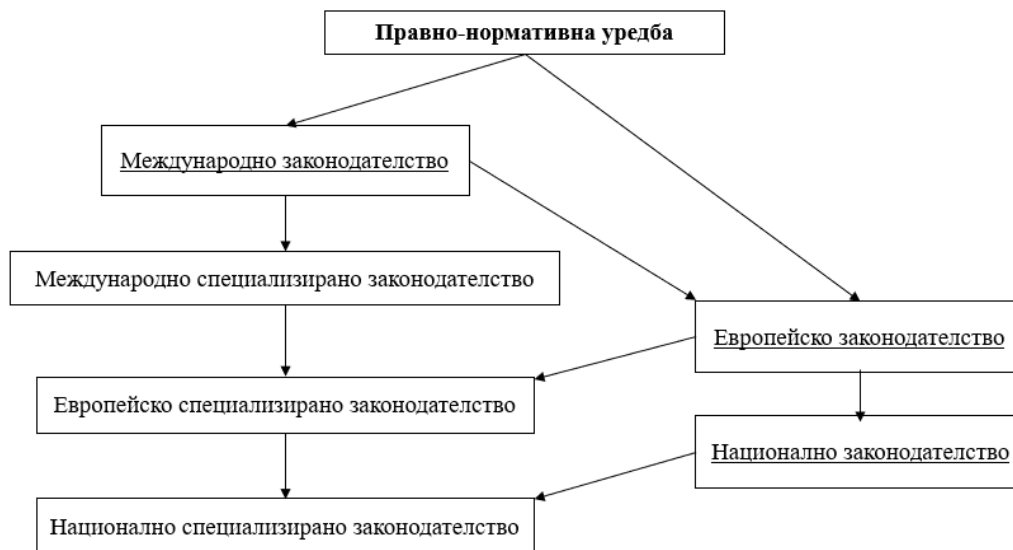
### **2.1. Актуални обстоятелства**

Насочването на усилията към разбираемо и същевременно полезно за практиката структуриране кореспондира с предварително задаване на константни критерии. Относно тематиката на настоящото изследване и съставните задачи, те са относими към комуникационното и информационно осигуряване (фигура 1). Като два основни признака за конкретизиране съответно са посочени нивото на политико-географско ориентиране и принадлежност по специализация. Именно поради това се представя обективната реалност, формираща асоциативни и кумулативни взаимовръзки, както и йерархични или линейни степени на зависимост. Обективността произлиза от практически потвърдените действия на ресорните международни и национални органи с юрисдикция на законни инстанции. Обстоятелството за реалистичност се основава на успешното и дългогодишното прилагане на такъв тип организиране на нормативи и стандарти. Предизвикателството за представянето им чрез модел с определени възможности е актуално и необходимо. По същество, стройното организиране на дадена дейност, особено когато тя засяга жизненоважни компоненти на обществени и икономически субекти, е отговорност с публична рефлексия. В конкретния случай се визира комуникационно и информационно осигуряване поради факта, че се обхващат процесите за придобиване на определен вид информация, която се привежда в подходяща форма, предава се чрез възможностите на комуникационните системи и се представя за използване на потребители с визуализиране в разбираем за тях формат. Правилата са дефинирани като правно-нормативна уредба, защото от една страна са със статут на правни норми за законосъобразност и в същото време съдържат конкретни нормативи (определени количества, които трябва да се постигнат от отделни единици).

Международното право е система от правни норми - договорни, обичайни и други, които регулират международните отношения в сферата на междувластни отношения [7].

Относно използването на термина законодателство, категорично се има предвид, че става въпрос за категории с императивен (задължителен) характер. Изпълнението им предполага не само избягване на нееднозначно тълкуване или противоречия, но и предвидимост на последствията във всеки един аспект. При моделирането на структурните решения това може да бъде зададено като отделен критерий, в случай че се предвижда възможност за използване и на документация с отворени стандарти, свободни коментари и т.н. Пряката връзка на структурирането с предназначението и свойствата на множествата се състои във взаимовръзките, които са показани. Конкретен пример може да бъде даден с това, че Европейското специализирано законодателство е едновременно част (подмножество) в обхвата на Световното специализирано законодателство и Европейското законодателство. Във връзка с това промяната на определени норми в Световното специализирано законодателство задължително следва да се отрази върху функционирането на Европейското специализирано законодателство, ако засягат Европа. В случай че промените се отнасят за

други региони на юрисдикция, действия по изменения и (или) допълнения в Европейското специализирано законодателство не би трябвало да са необходими.



**Фиг. 1.** Правна рамка на законодателството за комуникационно и информационно осигуряване

Аналогични са разсъжденията при взаимовръзката между Европейското законодателство и Европейското специализирано законодателство - промяна на определени норми в Европейски мащаб ще рефлектират, само ако засягат строго специализирана правна материя или стандарти. Важно е да се уточни, че е възможен вариант (предвиден във фигура 1), при който правоприлагащо на най-горно ниво е Европейското законодателство. Така се обосновава и необходимостта от решаването на генералния въпрос за синхронизиране на действието на правно-нормативната уредба. Разгледаните примери категорично са в презумпция за императивност и нива на подчиненост в аспект на глобална употреба. Това пряко кореспондира с конструирането на йерархична система за причинно-следствени взаимовръзки. При обстоятелства, касаещи предложения и актуализиране на нормативната база в ограничен регионален или корпоративен мащаб за частна употреба, е възможно прилагането не на йерархична, а на линейно-организирана система.

## 2.2. Практическа целесъобразност

След изясняване на същността и целите на структурирането, синтезът на подходящ модел може да се реализира с прилагането на ефективни инструменти. Крайните резултати следва да спомогнат постигането на максимална ефективност при лимитирани разходи на време и ресурси. Изграждането на цялостното решение изисква както използването на математически и логически операции, така и възможността те да бъдат материализирани с продуктови приложения. При избор на вариант за планиране и реализиране на определен модел е препоръчително да се предвидят няколко етапа, които да осигурят предвиждането на постепенната еволюция на реално съществуващите изходни данни и условия. По този начин моделът ще позволява адаптивност в динамична среда и използването на вложени ресурси на по-ранна фаза.

## 3. Системни архитектурни платформи и инструменти за структурно моделиране

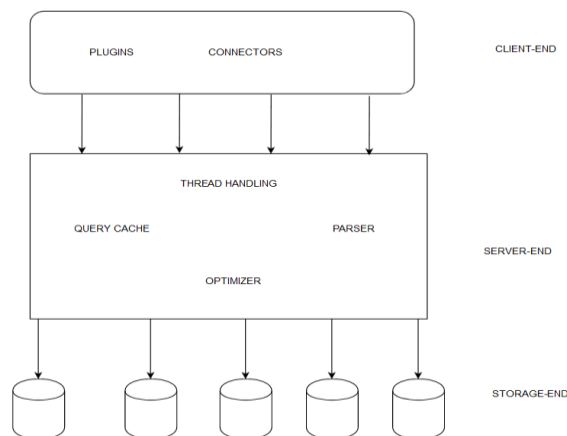
Разсъжденията по широката гама от възможности в това направление могат да се ориентират спрямо реалистични алгоритмични реализации и достъп до актуална и достоверна информация. Поради предявените принципни изисквания за актуалност и иновативен подход в работата, се налага базиране на достиженията на информационните технологии. Използването на решения от тази област на съвременното може да е едновременно

и достатъчно работоспособно, и отговарящо на повишеното ниво на очаквания от потребителите на информация. Предполага се, че последните са в състояние на нарастваща прогресия по информационна грамотност и в голямата си част разполагат с продуктивни персонални цифрови устройства. Имайки предвид и гореизложените подробности по определенията, с които се аргументира осъществяването на модел и структуриране на тематичните съдържания, подходящ инструментариум са базите от данни и възможностите за тяхното управление. Базата от данни представлява организирана колекция от структурирана информация или данни, обикновено съхранявана електронно в компютърна система. Базата от данни обикновено се контролира чрез система за управление на база от данни (СУБД). Данните и СУБД, както и приложенията, свързани с тях, заедно се наричат система за база от данни, често наричана за по-кратко просто база от данни [8]. Например определението, че по-правилно е всеки ред да се нарича „запис“, а всяка колона „поле“, като записът е смислен и последователен начин за обединяване на информацията за нещо, а полето е отделен информационен елемент - тип елемент, който присъства във всеки запис [9], [10] е безспорно и утвърдено. За максимална разбираемост се налага описанието, че във всеки един ред се дефинира даден обект, а в колоните, свързани с този ред, се дефинират определени негови отличителни свойства (характеристики). От друга страна съществуват противоречиви дефиниции като „Microsoft Access организира информацията в таблици: списъци с редове и колони. В простата база от данни може да има само една таблица. За повечето бази от данни трябва повече от една“ [9] и „Базите от данни и електронните таблици (например Microsoft Excel) са удобни начини за съхраняване на информация“. Базите от данни позволяват множество потребители едновременно да достъпват и заявяват данни бързо и защитено, използвайки изключително сложна логика и език [8]. Кореспондира ли въпросът „Каква е връзката на изложените дотук обяснения с мотивите за разработването на модел с йерархична структура?“. Отговорът е категоричен и положителен със следните аргументи:

- навигационните бази от данни, като йерархични такива използват „дървовидна“ структура, но позволяват единствено взаимовръзка „мултикаст“;
- мрежовите бази от данни са с повече възможности в сравнение с навигационните и могат да осъществят няколко взаимовръзки;
- навигационните и мрежовите бази от данни са опростени, но не достатъчно гъвкави;
- релационните бази от данни са ефективни, защото всеки един техен елемент е множество от таблици, но пък са задължителни правилата за съставянето им;
- обектно ориентираните бази от данни се основават на елементи, подходящи за съответните инструменти за програмиране;
- разпределените бази от данни са решение за резервираност като място за съхранение и налична комуникация.

Облачните бази от данни и самоадминистриращи се бази от данни са неизбежно съпроводени с лансиране на новите технологии. Например, Oracle Autonomous JSON Database е услуга за облачна база от данни с документи, която улеснява разработването на JSON-ориентирани приложения. Той включва API за документи в стил NoSQL (Oracle SODA и Oracle Database API за MongoDB), мащабиране без сървър, ACID транзакции с висока производителност, всеобхватна сигурност и ниски цени за плащане при използване. Автономната JSON база от данни автоматизира осигуряването, конфигурирането, настройката, мащабирането, корекцията, криптирането и поправката на бази от данни, елиминирайки управлението на базата от данни и предоставяйки 99,95% наличност [11]. Софтуерът на базата от данни улеснява управлението на данните, като дава възможност потребителите да съхраняват данни в структурирана форма и след това да ги достъпват [8]. За практическото прилагане на инструментариума и оползотворяване на възможностите в моделирането за йерархична структура е целесъобразно да се използва релационен тип база

от данни. По този начин ще се удовлетворяват изискванията за обработка на множество заявки и брой транзакции между отделни съдържания. Аргументирането се заключава в това, че определен нормативен документ може да принадлежи към различни дефинирани множества (например - отворен за коментари и допълнения международен стандарт, имащ отношение към специализирана насоченост), към който от една страна ще се позволи структуриране, а от друга ще се осигури непрекъснат многоточков достъп с регистриране на време от последна актуализация. Подходящо решение в този случай е езикът за структурирани запитвания - Structured Query Language (SQL)/MySQL. MySQL е система с отворен код за управление на релационна база от данни, базирана на SQL. Тя е проектирана и оптимизирана за WEB приложения и може да работи с всякакви платформи. С възникването на нови и различни изисквания във връзка с Интернет, MySQL става предпочитана платформа за WEB разработчиците и WEB базираните приложения [8]. От технологична гледна точка много важен елемент на цялостната структура е използването на организационна процедура „сървър – клиент“. Архитектурата на базата от данни обяснява връзката и взаимодействието между клиентската част, сървърната част и частта на системата за съхранение. По-долу са различните слоеве на системата от бази от данни [12]. На показаната схематична постройка за взаимодействие в MySQL (фигура 2), потребителите на информация използват клиентски интерфейс за достъп до базата от данни, асоцииран с клиентска част (CLIENT-END).



**Фиг. 2.** MySQL архитектура

В зависимост от конкретните предпочитания за работа е възможно графично визуализиране или такова чрез „команден ред“. Графичният потребителски интерфейс Graphical User Interface (GUI) е система от интерактивни визуални компоненти за компютърен софтуер. GUI показва обекти, които предават информация и представляват действия, които потребителят може да предприеме. Обектите променят цвета, размера или видимостта си, когато потребителят взаимодейства с тях [13]. Интерфейсът на „команден ред“ (Command Line Interface (CLI) е софтуерен механизъм, който използва за взаимодействие към операционната система помощта на клавиатура [14]. Формата на инициране на комуникацията между клиента и сървъра се проявява във вид на „заявка“.

#### **4. Материализиране на йерархичния модел за нормативната уредба с управление и разделяне в база от данни**

Процедурите следва да се съобразят с точно определена последователност от дейности. С оглед спазването на универсален подход, постигане на разбираемост и достъпност, може да се използва SQL с прилежащите му възможности. Задължително е да се изпълнят фундаментални целеви процеси. Необходимо е също да се изпълни и инструмент, който да свърже данните от таблиците. В обобщаващ план е правилно информацията да се разпредели



в разумен брой свързани таблици, предполагащ по-малко повтарящи се данни и по-малко места за редактиране. Акцентът е върху използването на SQL с данни, разпределени в няколко релационни таблици и събирането на данните от тях при необходимост. Прилаганата функция от синтаксиса се дефинира като правило, наречено „Join“.

Винаги трябва да се използват изрични съединения (Explicit joins). Неявните съединения (Implicit joins) имат няколко проблема [15]:

- условието за присъединяване е някъде в клаузата WHERE, смесено с всеки друг филтър условия. Това прави по-трудно да се види кои таблици са свързани и как.

- поради гореизложеното има по-висок риск от грешки и е по-вероятно те да бъдат открити по късно;

- в стандартния SQL изричните съединения са единственият начин за използване на външни съединения.

```
1 CREATE TABLE Regulations
2   (id INTEGER PRIMARY KEY AUTOINCREMENT,
3    name TEXT, year INTEGER, publisher TEXT,
4    standard TEXT);
5 INSERT INTO Regulations (name, year,
6 publisher, standard) VALUES ("ISO/IEC TR
7 11802-5", 1997, "International Organization
8 for Standardization", "Imperative");
9 INSERT INTO Regulations (name, year,
10 publisher, standard) VALUES ("ISO/IEC 29182
11 -1", 2013, "International Organization for
12 Standardization", "Imperative");
13 INSERT INTO Regulations (name, year,
14 publisher, standard) VALUES ("Software
15 -Defined Networking", 2012, "Open Networking
16 Foundation", "Open");
17 INSERT INTO Regulations (name, year,
18 publisher, standard) VALUES ("Request for
19 Comments 1392", 2013, "Internet Engineering
20 Task Force", "Open");
21 INSERT INTO Regulations (name, year,
22 publisher, standard) VALUES ("Request for
23 Comments: 3637", 2003, "Internet Engineering
24 Task Force", "Open");
25 INSERT INTO Regulations (name, year,
26 publisher, standard) VALUES ("ISO/IEC/IEEE
27 8802-3-1", 2015, "International Organization
28 for Standardization", "Imperative");
29 INSERT INTO Regulations (name, year,
30 publisher, standard) VALUES ("ISO/IEC/IEEE
31 8802-1AC", 2018, "International Organization
32 for Standardization", "Imperative");
33 INSERT INTO Regulations (name, year,
34 publisher, standard) VALUES ("Request for
35 Comments 3268", 2002, "Skygate Technology",
36 "Open");
```

Фиг. 3. Създаване на първа таблица с данни

За целите на изследванията е необходимо разглеждането на логическите взаимовръзки от фигура 1. Това ще подпомогне създаването на адекватно и полезно съдържание на релационните таблици. Във връзка в това трябва да се предвиди например, че определена норма с европейски обхват, ако е императивна (задължителна), следва да има връзка с такава на световно ниво. От друга страна, специализиран нормативен документ с европейски обхват може да е синхронизиран както с европейското общо законодателство, така и със световното

специализирано законодателство. В този ред на мисли разсъжденията могат да продължат и с другите визирани категории в низходящ ред.

Regulations	
id (PK)	INTEGER
name	TEXT
year	INTEGER
publisher	TEXT
standard	TEXT

Фиг. 4. Изглед на резултата за създадената таблица

```

21 CREATE TABLE Orientation_and_specification
22   (id INTEGER PRIMARY KEY AUTOINCREMENT,
23    name TEXT, area TEXT, part TEXT,
24    specification TEXT, theme TEXT);
25 INSERT INTO Orientation_and_specification
26   (name, area, part, specification, theme)
27   VALUES ("ISO/IEC TR 11802-5", "Telecommunicat
28   ions", "Media Access Control", "Local Area
29   Network", "Ethernet");
30
31 INSERT INTO Orientation_and_specification
32   (name, area, part, specification, theme)
33   VALUES ("ISO/IEC 29182-1", "Information
34   technology", "General overview", "Sensor
35   networks", "Sensor Network Reference
36   Architecture");
37
38 INSERT INTO Orientation_and_specification
39   (name, area, part, specification, theme)
40   VALUES ("Software-Defined Networking ",
41   "Telecommunications", "Technical report",
42   "White paper", " The new norm for networks");
43
44 INSERT INTO Orientation_and_specification
45   (name, area, part, specification, theme)
46   VALUES ("Request for Comments 1392", "Network
47   Working", "Technical report", "Users'
48   information", "Internet Users' Glossary");
49
50 INSERT INTO Orientation_and_specification
51   (name, area, part, specification, theme)
52   VALUES ("Request for Comments: 3637",
53   "Definitions of Managed Objects", "Interface
54   Sublayer", "Wide Area Network", "Ethernet");
55
56 INSERT INTO Orientation_and_specification
57   (name, area, part, specification, theme)
58   VALUES ("ISO/IEC/IEEE 8802-3-1", "Telecommuni
59   cations", "Standard for management informatio
60   n base", "Local Area Network", "Ethernet");
61
62 INSERT INTO Orientation_and_specification
63   (name, area, part, specification, theme)
64   VALUES ("ISO/IEC/IEEE 8802-1AC", "Telecommuni
65   cations", "Media Access Control", "Local Area
66   Network", "Service definition");
67
68 INSERT INTO Orientation_and_specification
69   (name, area, part, specification, theme)
70   VALUES ("Request for Comments 3268", "Network
71   Working", "Transport Layer", "Advanced
72   Encryption Standard", "Ciphersuites for
73   Transport Layer Security");

```

Фиг. 5. Създаване на втора таблица с данни

Regulations		Orientation_and_specification	
id (PK)	INTEGER	id (PK)	INTEGER
name	TEXT	name	TEXT
year	INTEGER	area	TEXT
publisher	TEXT	part	TEXT
standard	TEXT	specification	TEXT
		theme	TEXT

Фиг. 6. Общ изглед за резултата от създадените две таблици

```

40
41 SELECT * FROM Regulations, Orientation_and_sp
   ecification WHERE Regulations.id = Orientatio
   n_and_specification.id;
42
43 SELECT * FROM Regulations JOIN Orientation_an
   d_specification ON Regulations.id = Orientati
   on_and_specification.id;
44

```

Фиг. 7. Изпълнение на заявки с еднаква насоченост, но различен синтаксис, използващи неявно вътрешно присъединяване

id	name	year	publisher	standard	id	name	area	part	specification	theme
1	ISO/IEC TR 11802-5	1997	International Organization for Standardization	Imperative	1	ISO/IEC TR 11802-5	Telecommunications	Media Access Control	Local Area Network	Ethernet
2	ISO/IEC 29182-1	2013	International Organization for Standardization	Imperative	2	ISO/IEC 29182-1	Information technology	General overview	Sensor networks	Sensor Network Reference Architecture
3	Software-Defined Networking	2012	Open Networking Foundation	Open	3	Software-Defined Networking	Telecommunications	Technical report	White paper	The new norm for networks
4	Request for Comments 1392	2013	Internet Engineering Task Force	Open	4	Request for Comments 1392	Network Working	Technical report	Users' information	Internet Users' Glossary
5	Request for Comments: 3637	2003	Internet Engineering Task Force	Open	5	Request for Comments: 3637	Definitions of Managed Objects	Interface Sublayer	Wide Area Network	Ethernet
6	ISO/IEC/IEEE 8802-3-1	2015	International Organization for Standardization	Imperative	6	ISO/IEC/IEEE 8802-3-1	Telecommunications	Standard for management information base	Local Area Network	Ethernet
7	ISO/IEC/IEEE 8802-1AC	2018	International Organization for Standardization	Imperative	7	ISO/IEC/IEEE 8802-1AC	Telecommunications	Media Access Control	Local Area Network	Service definition
8	Request for Comments 3268	2002	Skygate Technology	Open	8	Request for Comments 3268	Network Working	Transport Layer	Advanced Encryption Standard	Ciphersuites for Transport Layer Security

Фиг. 8. Резултат от заявките с използването на неявно вътрешно присъединяване

```

45 SELECT Regulations.name, standard, specificat
   ion, theme FROM Regulations JOIN Orientation_
   and_specification ON Regulations.id =
   Orientation_and_specification.id;
46

```

Фиг. 9. Заявка за общ изглед с използване на „изрично вътрешно присъединяване“

name	standard	specification	theme
ISO/IEC TR 11802-5	Imperative	Local Area Network	Ethernet
ISO/IEC 29182-1	Imperative	Sensor networks	Sensor Network Reference Architecture
Software-Defined Networking	Open	White paper	The new norm for networks
Request for Comments 1392	Open	Users' information	Internet Users' Glossary
Request for Comments: 3637	Open	Wide Area Network	Ethernet
ISO/IEC/IEEE 8802-3-1	Imperative	Local Area Network	Ethernet
ISO/IEC/IEEE 8802-1AC	Imperative	Local Area Network	Service definition
Request for Comments 3268	Open	Advanced Encryption Standard	Ciphersuites for Transport Layer Security

Фиг. 10. Резултат от заявката, използвава изрично вътрешно присъединяване

```

47 SELECT Regulations.name, standard, specificat
ion, theme FROM Regulations JOIN Orientation_
and_specification ON Regulations.id =
Orientation_and_specification.id WHERE
standard = "Open" ORDER BY year;
48

```

Фиг. 11. Заявка с насоченост и използване на изрично вътрешно присъединяване

name	standard	specification	theme
Request for Comments 3268	Open	Advanced Encryption Standard	Ciphersuites for Transport Layer Security
Request for Comments: 3637	Open	Wide Area Network	Ethernet
Software-Defined Networking	Open	White paper	The new norm for networks
Request for Comments 1392	Open	Users' information	Internet Users' Glossary

Фиг. 12. Резултат от конкретизираната заявка, използвава изрично вътрешно присъединяване

След направените изследвания и потвърждението за ефективност по целесъобразност на използваните инструменти, е възможно предприемане на следващата стъпка с детализиране на йерархичната структура, показано на фигура 13.

Това позволява да се моделира и типова база от данни, включително:

- дефиниране документите в таблица за Регламенти (Regulations) и в таблица за насоченост и валидност (Targeting and validity);
- дефиниране на дванадесет типове документа (от Document A до Document L);
- фиксиране на характерните показатели на документите за принадлежност.

В настоящото изследване се предвижда структурирането на двете таблици по такъв начин, че да може да се извършва от различни длъжностни лица (екипи) по нива на компетентност.

За първата таблица е достатъчно да се познава типовият документ като наименование и дали е в обхвата на общото или специализираното законодателство (нормативна уредба). За тази цел не е необходимо да се познават подробно правните и технически детайли по взаимовръзките в йерархията на световното, европейско и национално законодателство.

За действия по създаването и попълването на втората таблица с данни е наложително да се приложат квалифицирани компетентности относно това даденият тип документ в каква юрисдикция е иницииран, от какво естество на обвързаност (статут) е тежестта му, приложим ли е на световно, европейско и национално ниво.



Фиг. 13. Детайлизирана структурна схема с йерархична обвързаност на нормативната уредба

Паралелно с това трябва да се съобрази и терминологията, използвана за задължителните или свободноизбираемите (отворени) стандарти. Разделянето на тези дейности е ефективно само при стриктно спазване на последователното индексирание (номерирание) на документите и техните наименования. Това са и единствените параметри по които ще се извърши присъединяването на таблиците една към друга. От практическа гледна точка е възможно, но не е разумно двете таблици да са релативни само по една колона (номер по ред или име).

```

1 CREATE TABLE Regulations
2   (id INTEGER PRIMARY KEY AUTOINCREMENT,
3    Name TEXT, General_legislation TEXT, Specialized_legislation
4    TEXT);
5 INSERT INTO Regulations (name, General_legislation, Specialized_1
6   egislation) VALUES ("Document A", "Yes", "No");
7 INSERT INTO Regulations (name, General_legislation, Specialized_1
8   egislation) VALUES ("Document B", "Yes", "No");
9 INSERT INTO Regulations (name, General_legislation, Specialized_1
10  egislation) VALUES ("Document C", "Yes", "No");
11 INSERT INTO Regulations (name, General_legislation, Specialized_1
12  egislation) VALUES ("Document D", "Yes", "No");
13 INSERT INTO Regulations (name, General_legislation, Specialized_1
14  egislation) VALUES ("Document E", "Yes", "No");
15 INSERT INTO Regulations (name, General_legislation, Specialized_1
16  egislation) VALUES ("Document F", "Yes", "No");
17 INSERT INTO Regulations (name, General_legislation, Specialized_1
18  egislation) VALUES ("Document G", "No", "Yes");
19 INSERT INTO Regulations (name, General_legislation, Specialized_1
20  egislation) VALUES ("Document H", "No", "Yes");
21 INSERT INTO Regulations (name, General_legislation, Specialized_1
22  egislation) VALUES ("Document I", "No", "Yes");
23 INSERT INTO Regulations (name, General_legislation, Specialized_1
24  egislation) VALUES ("Document J", "No", "Yes");
25 INSERT INTO Regulations (name, General_legislation, Specialized_1
26  egislation) VALUES ("Document K", "No", "Yes");
27 INSERT INTO Regulations (name, General_legislation, Specialized_1
28  egislation) VALUES ("Document L", "No", "Yes");

```

```

18 CREATE TABLE Targeting_and_validity
19     (id INTEGER PRIMARY KEY AUTOINCREMENT,
20      Name TEXT, Scope TEXT, Status TEXT, World_valid TEXT,
21      EU_valid TEXT, Bulgaria_valid TEXT);
22 INSERT INTO Targeting_and_validity (Name, Scope, Status,
23      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document A",
24      "World", "Binding", "Yes", "Yes", "Yes");
25 INSERT INTO Targeting_and_validity (Name, Scope, Status,
26      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document B",
27      "World", "Non-binding", "Yes", "By necessity", "By necessity");
28 INSERT INTO Targeting_and_validity (Name, Scope, Status,
29      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document C",
30      "Europe Union", "Regulatory", "On demand", "Yes", "Yes");
31 INSERT INTO Targeting_and_validity (Name, Scope, Status,
32      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document D",
33      "Europe Union", "Recommended", "No", "Yes", "By recommendation");
34 INSERT INTO Targeting_and_validity (Name, Scope, Status,
35      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document E",
36      "Bulgaria", "Mandatory", "No", "On demand", "Yes");
37 INSERT INTO Targeting_and_validity (Name, Scope, Status,
38      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document F",
39      "Bulgaria", "Declarative", "No", "No", "Yes");
40 INSERT INTO Targeting_and_validity (Name, Scope, Status,
41      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document G",
42      "World", "Standardizing", "Yes", "Yes", "Yes");
43 INSERT INTO Targeting_and_validity (Name, Scope, Status,
44      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document H",
45      "World", "Open", "Yes", "By necessity", "By necessity");
46 INSERT INTO Targeting_and_validity (Name, Scope, Status,
47      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document I",
48      "Europe Union", "Standard", "On demand", "Yes", "Yes");
49 INSERT INTO Targeting_and_validity (Name, Scope, Status,
50      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document J",
51      "Europe Union", "Opinion", "No", "Yes", "By recommendation");
52 INSERT INTO Targeting_and_validity (Name, Scope, Status,
53      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document K",
54      "Bulgaria", "State Standard", "No", "On demand", "Yes");
55 INSERT INTO Targeting_and_validity (Name, Scope, Status,
56      World_valid, EU_valid, Bulgaria_valid) VALUES ("Document L",
57      "Bulgaria", "Agreeable", "No", "No", "Yes");

```

Фиг. 14. Създаване на таблиците в базата от данни

При последващо селектиране на информация за тях е лесно, както да бъде филтрирано определено съдържание, така и да се изиска подредба в удобна за потребителя последователност. Това е демонстрирано на фигура 20, фигура 21, фигура 22 и фигура 23.

Regulations		Targeting_and_validity	
id (PK)	INTEGER	id (PK)	INTEGER
Name	TEXT	Name	TEXT
General_legislation	TEXT	Scope	TEXT
Specialized_legislation	TEXT	Status	TEXT
		World_valid	TEXT
		EU_valid	TEXT
		Bulgaria_valid	TEXT

Фиг. 15. Изглед на резултата за създадените таблици

```

35 SELECT * FROM Regulations, Targeting_and_validity WHERE
36      Regulations.id = Targeting_and_validity.id;

```

Фиг. 16. Заявка за визуализиране на обобщената информация

id	Name	General_legislation	Specialized_legislation	id	Name	Scope	Status	World_valid	EU_valid	Bulgaria_valid
1	Document A	Yes	No	1	Document A	World	Binding	Yes	Yes	Yes
2	Document B	Yes	No	2	Document B	World	Non-binding	Yes	By necessity	By necessity
3	Document C	Yes	No	3	Document C	Europe Union	Regulatory	On demand	Yes	Yes
4	Document D	Yes	No	4	Document D	Europe Union	Recommended	No	Yes	By recommendation
5	Document E	Yes	No	5	Document E	Bulgaria	Mandatory	No	On demand	Yes
6	Document F	Yes	No	6	Document F	Bulgaria	Declarative	No	No	Yes
7	Document G	No	Yes	7	Document G	World	Standardizing	Yes	Yes	Yes
8	Document H	No	Yes	8	Document H	World	Open	Yes	By necessity	By necessity
9	Document I	No	Yes	9	Document I	Europe Union	Standard	On demand	Yes	Yes
10	Document J	No	Yes	10	Document J	Europe Union	Opinion	No	Yes	By recommendation
11	Document K	No	Yes	11	Document K	Bulgaria	State Standard	No	On demand	Yes
12	Document L	No	Yes	12	Document L	Bulgaria	Agreeable	No	No	Yes

Фиг. 17. Изглед за обобщения резултат

```

37 SELECT Regulations.Name, General_legislation, Scope, Status
FROM Regulations JOIN Targeting_and_validity ON Regulations
.id = Targeting_and_validity.id;
38

```

Фиг. 18. Заявка за конкретизиран избор по определени критерии с използване на изрично вътрешно присъединяване по пореден номер

Name	General_legislation	Scope	Status
Document A	Yes	World	Binding
Document B	Yes	World	Non-binding
Document C	Yes	Europe Union	Regulatory
Document D	Yes	Europe Union	Recommended
Document E	Yes	Bulgaria	Mandatory
Document F	Yes	Bulgaria	Declarative
Document G	No	World	Standardizing
Document H	No	World	Open
Document I	No	Europe Union	Standard
Document J	No	Europe Union	Opinion
Document K	No	Bulgaria	State Standard
Document L	No	Bulgaria	Agreeable

Фиг. 19. Резултат от изпълнението на заявката

```

39 SELECT Targeting_and_validity.Name, World_valid, Bulgaria_v
alid FROM Regulations JOIN Targeting_and_validity ON
Regulations.id = Targeting_and_validity.id WHERE EU_valid =
"Yes" ORDER BY Scope;
40

```

Фиг. 20. Заявка за конкретизиран избор и азбучна подредба по обхват без визуализирането му с използване на изрично вътрешно присъединяване по пореден номер

За прецизиране на избора за нуждите от определена информация целево може да се наложат критерии, които да отсъстват или да се визуализират в „изхода на командата“. И двата варианта са осъществими, според желанието на потребителя.

Name	World_valid	Bulgaria_valid
Document C	On demand	Yes
Document D	No	By recommendation
Document I	On demand	Yes
Document J	No	By recommendation
Document A	Yes	Yes
Document G	Yes	Yes

**Фиг. 21.** Резултат от изпълнението на заявката

```

41 SELECT Regulations.id, Regulations.Name, Targeting_and_vali
    dity.Bulgaria_valid, Targeting_and_validity.Status FROM
    Regulations JOIN Targeting_and_validity ON Regulations.Name
    = Targeting_and_validity.Name WHERE EU_valid = "Yes" AND
    World_valid = "Yes" ORDER BY Regulations.id;
42

```

**Фиг. 22.** Заявка за конкретизиран избор и подредба по пореден номер с визуализиране и използване на изрично вътрешно присъединяване по име

id	Name	Bulgaria_valid	Status
1	Document A	Yes	Binding
7	Document G	Yes	Standardizing

**Фиг. 23.** Резултат от изпълнението на конкретизирания избор и подредба

Резултатът от изпълнението на последната детайлизирана заявка показва кои типове документи (в случая – А и G) са валидни едновременно на европейско и световно ниво, с какъв статут са (обвързващ и стандартизиращ) и че важат за изпълнение в България. С демонстрираните възможности категорично се онагледява способността на избраните инструменти за класифицирано селектиране на необходимата информация. Важно е да се отбележи и способността за следване на йерархичната системност в определени граници. Моделирането може да се надгради, като се създадат и допълнителни таблици. Една от тях например, може да приобщи не само конкретни заглавия и документи, принадлежащи към всеки един от типовете (от А до L), а и данни за тяхната спешност, дати на приемане и актуализация, обем и т.н. Практическата полза може особено да се засили при селектиране на документация по подготовка и извършване на проверки (одити), както и проектна активност.

## 5. Заключение

Настоящото емпирично изследване е терминологично аргументирано, притежава ясна практическа насоченост и налага изводите:

1. В условия на динамична по обхват и предназначение нормативна уредба за комуникационното и информационно осигуряване се налага търсенето и намирането на универсални решения с възможно по-широко приложение в практиката.

2. Демонстрираните инструменти за моделиране на йерархична структура са разбираеми, достъпни и ефективни.

3. В чисто специализиран инженерен аспект посочените възможности са приложими при конструкторски процедури, проучвания, експерименти и др. Бързите и изчерпателни справки на всеки един стадий от посочените дейности са индикация за компетентност и рейтингово позициониране.

## Литература

[1]. Соскова, А.А., Ефективна теория на моделите: Скок на структура, кодиране и декодиране – Автореферат // СУ „Св. Климент Охридски“, София, 2020.



- [2]. Мирчев, И., Графи. Оптимизационни алгоритми в мрежи, Университетско изд. „Неофит Рилски“, Благоевград, ISBN 954-680-174-7.
- [3]. Соскова, М., Записки по дискретна математика и алгоритми // СУ „Св. Климент Охридски“, София.
- [4]. Сариев, А.К., Аксиоматична теория на множествата // СУ „Св. Климент Охридски“, София, 2022.
- [5]. Касабов, И., Университетски речник - основни понятия // Нов Български университет, София, <https://nbu-rechnik.nbu.bg/bg/obsht-spisyk-na-ponqtiq/jerarhiq>, Последно посетен на 7.07.2024 г.
- [6]. Forward business, Йерархична и линейна организационна структура, Copyright Business Forward OOD, [https://www.b-forward.com/struktura\\_na\\_organizaciqta.html](https://www.b-forward.com/struktura_na_organizaciqta.html), Последно посетен на 8.07.2024 г.
- [7]. Дацов, В., Видин, Бл., Международно публично право, Тема №1 - Понятие за съвременно международно право. Определение. Характерни особености, София, <https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://vilidatsov.files.wordpress.com/2016/08/mpp-elektronen.pdf&ved=2ahUKEwiPkdPy15eHAxWrQvEDHYgfBjcQFnoECBMQAQ&usg=AOvVaw2DTj-g5C-qGqSaTq7sCJ06>, Последно посетен на 8.07.2024 г.
- [8]. Какво е база данни, Оракъл, <https://www.oracle.com/bg/database/what-is-database/>, Последно посетен на 05.06.2024 г.
- [9]. Основи на базите данни, Майкрософт, <https://support.microsoft.com/bg-bg/topic/%D0%BE%D1%81%D0%BD%D0%BE%D0%B2%D0%B8-%D0%BD%D0%B0-%D0%B1%D0%B0%D0%B7%D0%B8%D1%82%D0%B5-%D0%B4%D0%B0%D0%BD%D0%BD%D0%B8-a849ac16-07c7-4a31-9948-3c8c94a7c204>, Последно посетен на 04.06.2024 г.
- [10]. Базите данни - основа на всяка информационна система, Компютърно образователен център Progress, <https://progressbg.net/blog/database/bazite-danni-osnova-na-vsyaka-informacionna-sistema/>, Последно посетен на 04.06.2024 г.
- [11]. Autonomous JSON Database, Оракъл, <https://www.oracle.com/bg/autonomous-database/autonomous-json-database/>, Последно посетен на 06.06.2024 г.
- [12]. Achonwa Alvan, What is MYSQL architecture?, Educative License - Creative Commons-Attribution-ShareAlike 4.0 (CC-BY-SA 4.0), <https://www.educative.io/answers/what-is-mysql-architecture>, Последно посетен на 10.06.2024 г.
- [13]. What is GUI, Computer Hope, Интернет източник, Updated: 11/12/2023, <https://www.computerhope.com/jargon/g/gui.htm>, Последно посетен на 17.06.2024 г.
- [14]. What is CLI, Amazon Web Services (AWS), <https://aws.amazon.com/what-is/cli/>, Последно посетен на 17.06.2024 г.
- [15]. Stack Overflow contributors, Learning SQL, Free eBook, <https://riptutorial.com/>, Последно посетен на 17.06.2024 г.

**За контакти:**

инж. Георги С. Николов

Докторант в катедра „Комуникационна техника и технологии“

Технически университет - Варна

e-mail: [gnikolov@tu-varna.bg](mailto:gnikolov@tu-varna.bg)

# АВТОМАТИЗИРАН ПРОЦЕС ЗА РЕЗЕРВИРАНЕ ПРИ ИНТЕГРИРАН ОБМЕН НА СЕЛЕКТИРАНА ИНФОРМАЦИЯ

Георги Николов

**Резюме:** Статията разглежда подход за администриране, инсталиране и интегриране на мрежово оборудване с подходящите протоколи за обработка на информацията според типовете комуникация. За да се оползотворят възможностите на доставчиците на далекосъобщителни и комуникационни услуги, както и тези в локален план (които имат в болшинството случаи съответните финансово изражение и себестойност), квалифицираната аргументация е ключов фактор. Своевременното предвиждане на възможностите, които се предоставят, е предпоставка за пълноценно реализиране на работещи и практически полезни хардуерни и софтуерни конфигурации. Прецизното изпълнение на същите е много силен аргумент за удовлетворяване на изискването за гарантирана свързаност и непрекъснатост на достъпа до информация в дългосрочен период от време, включително чрез осигуряване на резервираност.

**Ключови думи:** Автоматизирано резервиране, мрежова комуникация, информация, Интернет, доставчици на Интернет услуги, топология, маршрутизатор, комутатор, маршрут, протокол за граничен шлюз, протокол за маршрутизатор в горещ режим на готовност, маршрутна таблица.

## Automated reservation process with integrated exchange of selected information

Georgi Nikolov

**Abstract:** The article examines an approach to the administration, installation and integration of network equipment with the appropriate protocols for processing information according to the types of communication. In order to exploit the possibilities of telecommunications and communication service providers, as well as those in the local plan (which in most cases have the corresponding financial expression and cost), qualified argumentation is a key factor. Timely anticipation of the opportunities that are provided is a prerequisite for the full implementation of working and practically useful hardware and software configurations. The precise implementation of the same is a very strong argument for satisfying the requirement for guaranteed connectivity and continuity of access to information over a long period of time, including by ensuring redundancy..

**Key words:** Automated redundancy, network communication, information, Internet, Internet Service Providers, topology, router, switch, route, Border Gateway Protocol, Hot Standby Router Protocol, routing table.

### 1. Въведение

Качеството на информационните услуги е мярката за комплексна оценка при предаване и наличността им в ограничена по обхват (LAN) или Интернет (WAN) среда. Наличността на конкретна услуга е важен и основен елемент в осигуряването на потребителите. Мрежовата инфраструктура трябва да бъде проектирана така, че да бъде пределно достъпна, преди да може успешно да се внедри. Целта за висока наличност е максимално време на работа, като се допускат само „няколко минути престой на година“. Комуникационните мрежи формират основата на техническите платформи във всяка успешна съвременна организация. Тези мрежи гарантират обмен на информация за множество приложения, включително глас в реално време, висококачествено видео и чувствителни към забавяне данни. Мрежовите инфраструктури трябва да предоставят предвидими, измерими и в болшинството случаи гарантирани услуги чрез управлението и контрола на пропускателна способност, забавяне и

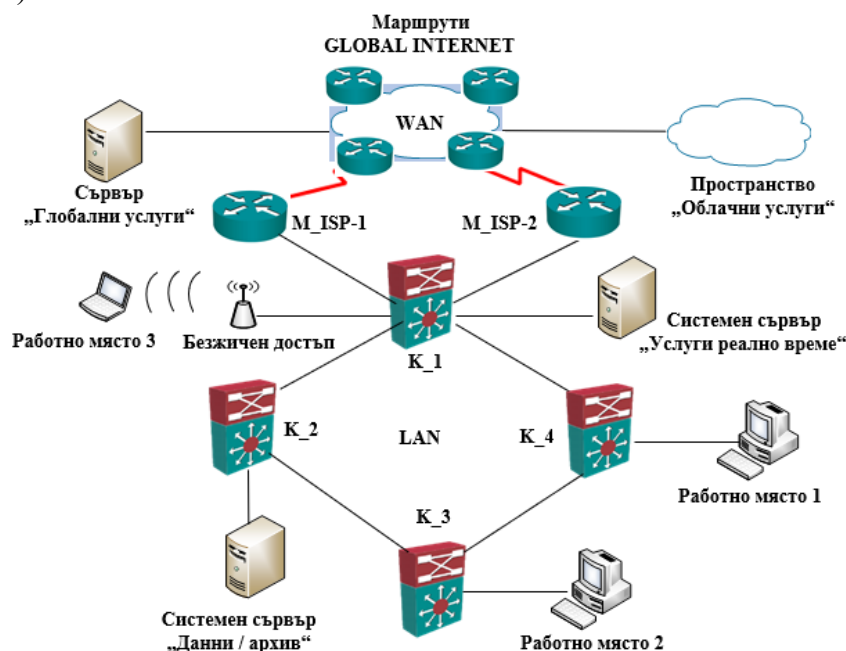
поносим толеранс от евентуална загуба на информация (данни). Изборът на технологични решения се отнася до набор от инструменти и механизми за мениджмънт на мрежовите ресурси и се счита за ключово определяща съставна на мрежовата конвергенция.

Приложенията за глас, видео, публични или чувствителни данни могат да получат приоритетна или преференциална обработка от мрежовите устройства, така че качеството на тези приложения да не се влоши до точката на неизползваемост. По същество основно и структуроопределящо мероприятие в организационен и технически аспект е резервирането на комуникациите. Целта е постижима с осигуряване на варианти за маршрутизиране на информационен трафик, без да се изпада в крайна зависимост от работоспособността на отделен маршрутизатор или комутатор в топологията на мрежата.

## 2. Автоматизирано резервиране на мрежова комуникация при обмен на селектирана информация

### 2.1. Актуални обстоятелства в практически целесъобразна и приложима топология

Комуникацията и много аспекти на живота, които включват комуникация, са силно повлияни от Интернет. Интернет позволява информацията да се обменя между ползвателите и/или субектите бързо и относително лесно. Интернет включва много мрежи, възли, които са свързани заедно, така че информацията да може да се разпределя и прехвърля между тях. Някои мрежови възли може да са маршрутизатори, които разпространяват пакети от една връзка към друга, могат да бъдат индивидуални клиентски компютри, може да са лични мрежи за различни обекти (напр. Интранет за фирми) и т.н. [1]. В обхвата на проведеното изследване се разглеждат функционалности с практически насочена реализация за резервиран обмен на информация в граничещата с Интернет среда в локален и публичен достъп (фигура 1).



Фиг. 1. Топология на реална комуникационна среда с интегриране на услугите

Приоритет в изходното задание и емпиричната постановка е резервирането на интегрирания обмен на селектирана информация. Същият е необходим при взаимодействие на организация или учреждение с комуникационната среда на системни интегратори и

доставчици на услуги, като информация с различно естество и предназначение трябва да преминава от една точка на мрежовата архитектура до друга с висока степен на достоверност и непрекъснатост. С особена важност това се отнася за случаите по възникване на прекъсвания и откази на изделия и/или връзки между тях. На фигура 1 са показани комуникационно-информационни и реално съществуващи елементи от глобалната среда (Сървър „Глобални услуги“, маршрути на глобалния Интернет и пространство за „Облачни услуги“). На следващо място са изделията на доставчиците на Интернет услуги (Internet Service Providers) като същите предоставят съответната свързаност до граничните маршрутизатори (M\_ISP-1 и M\_ISP-2), които като администраторски контрол и конфигуриране са ключови за ведомствено-мрежовата инфраструктура. Останалата част от топологията представя характерните информационни ресурси и видове комуникации в локалната среда. Основната част за осигуряване на комуникацията е мрежата от комутатори (K\_1, K\_2, K\_3 и K\_4), системните сървъри за минимално изискуемите услуги в реално време (Domain Name Service - DNS, World Wide Web - www и т.н), данни и архив. Особено важно е да се спомене, че интегрираният подход с краен резултат и резервирането са необходими, за да позволят функционалността на всяка една услуга, независимо от нейното естество. Задължително е съобразяването с факта, че в редица организации информацията между локалната и глобална среда периодично се синхронизира. За целта е необходима непрекъснатата комуникация между локално наличния и Интернет базирания и/или виртуализиран ресурс. При научни изследвания изпълнението на основни международни стандарти е солидна аргументация по изпълнение на изискванията за универсалност и съвместимост. Резултатите от радиолокационно наблюдение и радионавигационно осигуряване и кореспондиращата с тях информация за метеорологичната обстановка в даден район са чувствителни по показател за своєвременност. Пряко свързан е стандартът за телекомуникациите и обмена на информация между системи за информационни технологии с Изисквания за локални и градски мрежи – „Локални и градски мрежи - Част 1AS: Време и синхронизация за чувствителни към времето приложения в мостови локални мрежи“. Този документ определя протоколи, процедури и управлявани обекти, използвани за гарантиране, че изискванията за синхронизация са изпълнени за чувствителни към времето приложения, като аудио, видео и чувствителен към времето контрол в мрежи, например IEEE 802 и подобни комуникационни среди. Това включва синхронизация по време при нормална работа и след добавяне, премахване или отказ на мрежови компоненти и преконфигуриране на мрежата [2]. В резултат от функционалността на реализираната свързаност на локалната и глобална среда е осигуряване на потребителите за показаните на фигура 1 работни станции. Налага се ISO/IEC/IEEE 8802-1AX:2021. Този стандарт използва следните термини, дефинирани в IEEE Std 802: крайна станция, логическа връзка, станция (сървър). Този стандарт използва следните термини, дефинирани в IEEE Std 802.1AC: клиент (7.1 от IEEE Std 802.1AC-2016), рамка (7.2 от IEEE Std 802.1AC-2016), вътрешна услуга на подслоя (ISS) (кляуза 11 от IEEE Std 802.1AC-2016) - порт (7.4 от IEEE Std 802.1AC-2016) [3].

В проведените изследвания в локалната среда се използва свързаност, базирана на „Ethernet“ протокол. Точното изпълнение на ISO/IEC/IEEE 8802-3:2021 по изискванията за локални и градски мрежи - Част 3: Стандарт за Ethernet е задължително. Този документ дефинира Ethernet локални мрежи, мрежи за достъп и градски мрежи. Ethernet се определя при избрани скорости на работа и използва обща спецификация за контрол на достъпа до комуникационната среда (MAC) и информационна база за управление (MIB). MAC-протоколът Carrier Sense Multiple Access with Collision Detection (CSMA/CD) определя операция на споделен носител (полудуплекс), както и работа на пълен дуплекс. Специфичните скорости за независими медийни интерфейси (MII) осигуряват архитектурен и незадължителен интерфейс за избрани обекти на физическия слой (PHY). Той кодира кадри за предаване и декодира получените кадри с модулацията, определена за скоростта на

работа, предавателната среда и поддържаната дължина на връзката [4]. Принципно, в направените изследвания и обектите на интерес в комуникационно-информационната среда са съобразени с работа в мрежа, включително локални мрежи (LAN), градски мрежи (MAN), широкообхватни мрежи (WAN) и други [5]. С поетапно разглежданите функционалности в последователен логически ред е възможно и надграждането в аргументацията, управлението и уточненията, които формулират задачи за по-нататъшни решения чрез реализиране на софтуерно дефинирана мрежа (SDN) като се даде възможност за разработване на мрежи от следващо поколение. SDN подходът включва отделяне на препращането и контролни равнини, така че да се извършва контрол на мрежата посредством логически-централизирана контролна равнина, която се намира във външен елемент и програмира равнината за пренасочване с използване на отворени интерфейси. Сред предимствата, въведени от SDN, е способността да се контролира мрежата, като се вземе предвид цялостното мрежово състояние, което е от интерес за мрежовите оператори. Тази способност на SDN може да революционизира инженеринга на трафика (TE) [6].

## 2.2. Реализиране на топология и осъществяване на резервираност

**2.2.1.** По отношение на комуникацията между устройствата в публичната среда, включително към/от граничните маршрутизатори се използват възможните протоколи за серийни линии, описани в RFC 1392 [7] или спецификата по изграждане на „Ethernet“ за WAN Interface Sublayer [8]. Във вътрешния сегмент се използва „Ethernet“ протокол като приложим и стандартен метод за окомплектоване на Интернет Протокол (IP)-„дейтаграми“ в „Ethernet“. Функционалността и критериите за приложимост се основават на това, че IP-„дейтаграмите“ се предават в стандартни „Ethernet“ фреймове. Преобразуването на адреси, статична таблица, динамично откриване, идентификация на адрес за излъчване и форматите на трейлъри [9] са възможности за относимост и отношение към съдържателност в представянето на комплектуваните IP-пакети. В съответстващия тип рамка целите по резервирането и интеграцията обмен на селектирана информация се удовлетворяват.

**2.2.2.** За целите на емпиричните изследвания и реализиране на процесите е използвано типова комуникационно оборудване в експлоатационна поддръжка от производителя:

- Модел маршрутизатор - Cisco® 4331 Integrated Services Routers (фигура 2), принадлежащ към серията Cisco 4000 Family Integrated Services Router (ISR), която революционизира Wide Area Network (WAN) комуникациите в корпоративния клон.

С нови нива на вградени интелигентни мрежови възможности и конвергенция, той специално адресира нарастващата нужда от мрежи, съобразени с приложенията, в разпределени корпоративни сайтове. Тези местоположения обикновено разполагат с икономични ИТ ресурси. Но те често имат и нарастваща нужда от директна комуникация както с частни центрове за данни, така и с публични облаци през различни връзки, включително Virtual Private Network (VPN) с многопротоколно превключване на етикети (Multi-Protocol Label Switching - MPLS) и Интернет. Серията „Cisco 4000“ съдържа следните платформи: 4461, 4451, 4431, 4351, 4331, 4321 и 4221 ISR [10].



**Фиг. 2.** Маршрутизатор от серията Cisco® 4000 Integrated Services Routers, източник: [10]

Cisco обявява датите за край на продажбата и края на живота на Cisco ISR4200, ISR4300 и избрана платформа от серията ISR4400. Клиентите с активни договори за услуги ще продължат да получават поддръжка от Центъра за техническа помощ на Cisco (TAC), както е показано в Таблица 1 на End of Life (EoL) бюлетина. Таблица 1 описва етапите на изтичане на жизнения цикъл, дефинициите и датите за засегнатия продукт(и) [11].

**Таблица 1.** Съобщение за края на продажбата и края на експлоатацията на Cisco ISR4200, ISR4300 и избрана платформа от серия ISR4400 [11]

Понятие	Определение	Дата
Дата на края на изданията за поддръжка на софтуера:	Последната дата, на която Cisco Engineering може да пусне всякакви окончателни версии за поддръжка на софтуер или корекции на грешки. След тази дата Cisco Engineering повече няма да разработва, ремонтира, поддържа или тества софтуера на продукта.	31 август 2025 г.
Край на поддръжката за уязвимост/сигурност:	Последната дата, на която Cisco Engineering може да пусне планирано издание за поддръжка или планирано софтуерно решение за проблем с уязвимост на сигурността.	30 ноември 2028 г.
Дата на края на рутинния анализ на повредата:	Последната възможна дата, на която може да се извърши рутинен анализ на повреда, за да се определи причината за повреда или дефект на хардуерен продукт.	6 ноември 2024 г.
Крайна дата на подновяване на договора за услуга:	Последната дата за удължаване или подновяване на договор за услуга за продукта.	5 февруари 2028 г.
Последна дата на поддръжка:	Последната дата за получаване на приложимото обслужване и поддръжка за продукта, съгласно действащите договори за обслужване или съгласно гаранционните условия. След тази дата всички услуги за поддръжка за продукта са недостъпни и продуктът става остарял.	30 ноември 2028 г.

```

Router>show version
Cisco IOS Software [Everest], ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version 16.6.4,RELEASE SOFTWARE (fc3)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Sun 08-Jul-18 04:33 by mcpre

Cisco IOS-XE software, Copyright (c) 2005-2018 by cisco Systems, Inc.
All rights reserved. Certain components of Cisco IOS-XE software are
licensed under the GNU General Public License ("GPL") Version 2.0. The
software code licensed under GPL Version 2.0 is free software that comes
with ABSOLUTELY NO WARRANTY. You can redistribute and/or modify such
GPL code under the terms of GPL Version 2.0. For more details, see the
documentation or "License Notice" file accompanying the IOS-XE software,
or the applicable URL provided on the flyer accompanying the IOS-XE
software.

cisco ISR4331/K9 (1RU) processor with 1795999K/6147K bytes of memory.
Processor board ID FLM232010G0
3 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
3207167K bytes of flash memory at bootflash:.

OK bytes of WebUI ODM Files at webui:.

```

**Фиг. 3.** Информация за използваната версия на маршрутизаторите

- Модел комутатор - WS-C3650-24PS (фигура 4). Серията Cisco® Catalyst® 3650 е следващото поколение самостоятелни и подражаеми комутатори за ниво на достъп от корпоративен клас, които осигуряват основата за пълна конвергенция между кабелни и безжични на една платформа. Серията 3650 е изградена върху усъвършенствания Cisco StackWise®-160 и се възползва от новата специфична за приложението интегрална схема (ASIC) на Cisco Unified Access™ Data Plane (UADP). Този комутатор може да даде

възможност за прилагане на единна политика за жични и безжични мрежи, видимост на приложенията, гъвкавост, оптимизиране на приложенията и превъзходна устойчивост. Комутаторите от серия 3650 поддържат пълен IEEE 802.3at Power over Ethernet Plus (PoE+), Cisco Universal Power over Ethernet (Cisco UPOE®) на мултигигабитовите комутатори от серия Cisco Catalyst 3650 и предлагат модулни и сменяеми на място резервни вентилатори и захранвания [12].



**Фиг. 4.** Комутатори от серия Cisco Catalyst 3650, източник: [12]

Cisco обявява датите за край на продажбата и края на жизнения цикъл на Cisco Catalyst 3650. Клиентите с активни договори за услуги ще продължат да получават поддръжка от Център за техническа помощ на Cisco (TAC), както е показано в таблица 2 на бюлетина EoL. Таблица 2 описва етапите на изтичане на жизнения цикъл, дефинициите и датите за засегнатия продукт(и). За клиенти с активни и платени договори за обслужване и поддръжка, поддръжката ще бъде достъпна съгласно условията на договора за обслужване на клиента [13].

**Таблица 2.** Съобщение за края на актуалност по продажбата и края на експлоатацията на Cisco Catalyst 3650 [13]

Понятие	Определение	Дата
Край на поддръжката за уязвимост/сигурност:	Последната дата, на която Cisco Engineering може да пусне планирано издание за поддръжка или планирано софтуерно решение за проблем с уязвимост на сигурността.	31 октомври 2026 г.
Крайна дата на подновяване на договора за услуга:	Последната дата за удължаване или подновяване на договор за услуга за продукта.	29 януари 2026 г.
Последна дата на поддръжка:	Последната дата за получаване на приложимото обслужване и поддръжка за продукта, съгласно действащите договори за обслужване или съгласно гаранционните условия. След тази дата всички услуги за поддръжка за продукта са недостъпни и продуктът става остарял.	31 октомври 2026 г.

```
Switch>show version
Cisco IOS Software [Denali], Catalyst L3 Switch Software (CAT3K_CAA-UNIVERSALK9-M), Version 16.3.2, RELEASE SOFTWARE (fc4)
Technical Support : http://www.cisco.com/techsupport
Copyright(c) 1986 - 2016 by Cisco Systems, Inc.
Compiled Tue 08 - Nov - 16 17:31 by pt_team

cisco WS-C3650-24PS (MIPS) processor (revision N0) with 865815K/6147K bytes of memory.
Processor board ID FDO2031E1Q6
1 Virtual Ethernet interface
28 Gigabit Ethernet/IEEE 802.3 interface(s)
2048K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
250456K bytes of Crash Files at crashinfo : .
1609272K bytes of Flash at flash : .
OK bytes of at webui : .
```

**Фиг. 5.** Информация за използваната версия на комутаторите

**2.2.3.** В проведените изследвания за мрежов комуникационен сегмент по отношение на публична свързаност се използват два независими доставчика на услуги и Интернет (ISP). Това е необходимото условие за осъществяване на резервираност в редица организации с подходяща кореспонденция и еднозначно уточнение относно определеното адресно пространство и/или автономна система с протоколиране на екстериорната среда. За високо ниво на приложимост на резултатите от емпиричната постановка и реална ефективност се

използва Border Gateway Protocol (BGP). Това е протокол за маршрутизиране между автономни системи. Основната функция на BGP комуникаращата система е да обменя информация за достъпността на мрежата с други BGP системи. Тази информация за достъпност на мрежата включва и данни от списъка с автономни системи, през които преминава комуникацията. Тази информация е достатъчна за конструиране на графика на свързаността на автономните системи за тази достъпност, от която могат да бъдат елиминирани вериги за маршрутизиране и могат да бъдат наложени някои решения за мрежови политики. BGP-4 предоставя набор от механизми за поддръжка на безкласово маршрутизиране между домейни (CIDR). Тези механизми включват поддръжка за обявяване на набор от дестинации като IP-префикс и елиминирание на концепцията за мрежов клас в BGP. BGP-4 също въвежда механизми, които позволяват агрегиране на маршрути, включително агрегиране на пътища за автономни системи [14]. За осъществяване на съответната комуникация, включително с обявяване на определена вътрешноведомствена публична мрежа или сегмент, маршрутизаторите M\_ISP-1 и M\_ISP-2 са конфигурирани както следва:

```
M_ISP-1#show running-config
Building configuration...

Current configuration : 2168 bytes
!
version 15.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname M_ISP-1
interface Serial0/2/0
 ip address 46.10.149.19 255.255.255.248
 ip nat outside

router bgp 65001
 bgp router-id 1.1.1.1
 bgp log-neighbor-changes
 no synchronization
 neighbor 46.10.149.17 remote-as 8866
 network 78.83.151.0 mask 255.255.255.0
```

**Фиг. 6.** Информация за конфигурация на маршрутизатор M\_ISP-1, осигуряваща BGP

```
M_ISP-2#show running-config
Building configuration...

Current configuration : 2125 bytes
!
version 15.4
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname M_ISP-2
interface Serial0/2/1
 ip address 78.83.15.131 255.255.255.248
 ip nat outside

router bgp 65002
 bgp router-id 2.2.2.2
 bgp log-neighbor-changes
 no synchronization
 distance 70 78.83.15.129 0.0.0.0
 neighbor 78.83.15.129 remote-as 29580
 network 78.83.151.0 mask 255.255.255.0
```

**Фиг. 7.** Информация за конфигурация на маршрутизатор M\_ISP-2, осигуряваща BGP



Потвърждение за свързаността се доказва от съдържанието в маршрутните таблици на M\_ISP-1 и M\_ISP-2, насочено към публичен ресурс, представляващ интерес за достъп в реално време или константна за продължителен период информация:

```
M_ISP-1#show ip route | section B
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
B      64.246.161.0/24 [20/0] via 46.10.149.17, 00:00:00
       78.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
B      184.16.60.0/24 [20/0] via 46.10.149.17, 00:00:00

M_ISP-2#show ip route | section B
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
B      64.246.161.0/24 [70/0] via 78.83.15.129, 00:00:00
       78.0.0.0/8 is variably subnetted, 4 subnets, 3 masks
B      184.16.60.0/24 [70/0] via 78.83.15.129, 00:00:00
```

**Фиг. 8.** Информация за свързаността на маршрутизатори M\_ISP-1 и M\_ISP-2, насочена към селектиран публичен ресурс

При изградената топология единият от маршрутите е в режим на използване, а другият е „в резерв“ при прекъсване на връзка и/или отпадане на елемент от инфраструктурата. Кой маршрут е основен за изходящия трафик зависи от начина на конфигуриране с резервиращ протокол. По отношение на входящия трафик предпочитан е маршрутът с BGP през M\_ISP-1. Определя се от по-малката административна дистанция (със стойност 20, която е „по подразбиране“ за протокола) в сравнение с тази през M\_ISP-2 (където тя е със стойност 70 и е наложена целево). Предвижда се и транслирането на частно адресирани мрежи на организацията към публична адресация чрез записа „ip nat outside“ в конфигурацията на интерфейси Serial0/2/0 на M\_ISP-1 и Serial0/2/1 на M\_ISP-2 (фигура 7). Детайлните възможности са конкретизирани в RFC (Request for Comments) 7857, актуализиращи RFC 4787, RFC 5382 и RFC 5508.

**2.2.4.** Като конкретен механизъм (инструмент) за автоматично резервиране на комуникациите и маршрутизиращ протокол се използва Hot Standby Router Protocol (HSRP). Този протокол е един от начините за постигане на почти сто процента непрекъсната работа на мрежата, като се осигурява мрежово резервиране за IP мрежи и гарантиране, че потребителският трафик незабавно и прозрачно се възстановява от грешки при първи „хопове“ в крайните за мрежата устройства или вериги за достъп. Когато два или повече маршрутизатора споделят IP адрес и MAC (Layer 2) адрес, те могат да работят като един виртуален маршрутизатор. Членовете на групата „виртуален маршрутизатор“ непрекъснато обменят съобщения за статуса си. По този начин един маршрутизатор може да „поеме отговорността“ за маршрутизиране от друг, който не работи поради планирани или непланирани причини. Хостовете продължават да препращат IP пакети към постоянен IP и MAC адрес, а смяната на устройствата, които извършват маршрутизирането, е „прозрачна“ [15], [16]. Конфигурациите, касаещи използването на HSRP в M\_ISP-1 и M\_ISP-2, използват ключовите възможности от „standby ip“, „standby priority“ и „standby preempt“ за всяка една избрана група и е както следва:

```

M_ISP-1#show running-config
Building configuration...

Current configuration : 2001 bytes

interface GigabitEthernet0/0/0
no ip address
duplex auto
speed auto
!
interface GigabitEthernet0/0/0.10
encapsulation dot1Q 10
ip address 10.10.30.2 255.255.255.0
ip nat inside
standby 1 ip 10.10.30.1
standby 1 priority 120
standby 1 timers 10 60
standby preempt
!
interface GigabitEthernet0/0/0.99
encapsulation dot1Q 99
ip address 172.16.1.2 255.255.255.0
ip nat inside
standby 3 ip 172.16.1.1
standby 3 priority 120
standby 3 preempt
standby 3 timers 10 60
!
interface GigabitEthernet0/0/0.192
encapsulation dot1Q 192
ip address 192.168.30.2 255.255.255.0
ip nat inside
standby 2 ip 192.168.30.1
standby 2 priority 120
standby 2 preempt
standby 2 timers 10 60
!
interface GigabitEthernet0/0/0.200
encapsulation dot1Q 200
ip address 78.83.151.2 255.255.255.0
standby 4 ip 78.83.151.1
standby 4 preempt
standby 4 timers 10 60

M_ISP-2#show running-config
Building configuration...

Current configuration : 1978 bytes

interface GigabitEthernet0/0/0
no ip address
duplex auto
speed auto
!
interface GigabitEthernet0/0/1
no ip address
duplex auto
speed auto
!
interface GigabitEthernet0/0/1.10
encapsulation dot1Q 10
ip address 10.10.30.3 255.255.255.0
ip nat inside
standby 1 ip 10.10.30.1
standby 1 timers 10 60
standby preempt
!
interface GigabitEthernet0/0/1.99
encapsulation dot1Q 99
ip address 172.16.1.3 255.255.255.0
ip nat inside
standby 3 ip 172.16.1.1
standby 3 preempt
standby 3 timers 10 60
!
interface GigabitEthernet0/0/1.192
encapsulation dot1Q 192
ip address 192.168.30.3 255.255.255.0
ip nat inside
standby 2 ip 192.168.30.1
standby 2 preempt
standby 2 timers 10 60
!
interface GigabitEthernet0/0/1.200
encapsulation dot1Q 200
ip address 78.83.151.3 255.255.255.0
standby 4 ip 78.83.151.1
standby 4 priority 120
standby 4 preempt
standby 4 timers 10 60

```

**Фиг. 9.** Информация за осигуряване на HSRP от маршрутизатори M\_ISP-1 и M\_ISP-2

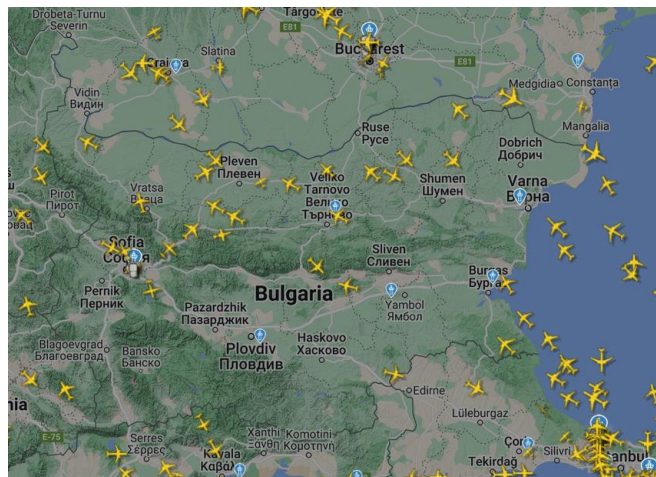
С изградената конфигурация изходящият трафик от мрежи 10.x.x.x, 172.16.x.x и 192.168.x.x ще се насочва през M\_ISP-1, а този от публичния сегмент - през M\_ISP-2. Постига се безспорна креативност с резервиране чрез „разминаване на маршрутите“, напомняйки, че приоритетно входящият публичен трафик се приема от M\_ISP-1. Виртуализацията с подинтерфейси върху реален физически порт (GigabitEthernet0/0/0), използването на VLANs (съгласно RFC 2674 [17]) и стволата „Trunk“ комуникация „маршрутизатор-комутатор“ е оползотворена функционалност на изделията Cisco 4331 ISR и Catalyst WS-C3650-24PS. Достъпността до публичния ресурс от радиолокационно наблюдение от работна станция със съответното визуализиране е показана на фигура 10, а по отношение на заложената информация от радионавигационно обслужване - на фигура 11.

```
C:\>ping 184.16.60.101

Pinging 184.16.60.101 with 32 bytes of data:

Reply from 184.16.60.101: bytes=32 time=1ms TTL=126
Reply from 184.16.60.101: bytes=32 time=5ms TTL=126
Reply from 184.16.60.101: bytes=32 time=1ms TTL=126
Reply from 184.16.60.101: bytes=32 time=1ms TTL=126

Ping statistics for 184.16.60.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 5ms, Average = 2ms
```



Фиг. 10. Информация за достъп до публичен ресурс с изменяща се в реално време обстановка

```
C:\>ping 64.246.161.213

Pinging 64.246.161.213 with 32 bytes of data:

Reply from 64.246.161.213: bytes=32 time=1ms TTL=126
Reply from 64.246.161.213: bytes=32 time=2ms TTL=126
Reply from 64.246.161.213: bytes=32 time=1ms TTL=126
Reply from 64.246.161.213: bytes=32 time=1ms TTL=126

Ping statistics for 64.246.161.213:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 2ms, Average = 1ms
```



Фиг. 11. Информация за достъп до публичен ресурс с константен статут

Положителният резултат е отправна точка за проверка на автоматичното резервиране при възникване на неблагоприятни обстоятелства по използване на основната комуникация.

### 2.2.5. Функционалност на резервирането при осигуряване на маршрутите

Доказва се чрез мониторинг при отказ на устройство и/или прекъсване на връзка.

При липса на отказ, приоритетно, желаната дестинация от избрана работна станция с публична адресация в изходяща посока към Интернет се осигурява от M\_ISP-2 (фигура 12).

```
C:\>tracert 184.16.60.101

Tracing route to 184.16.60.101 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    78.83.151.3
  1  1 ms    0 ms    0 ms    78.83.15.129
  2  1 ms    0 ms    0 ms    184.16.60.101

Trace complete.
```

**Фиг. 12.** Информация за трасето на комуникацията и функционалността по основен маршрут

При невъзможност маршрутизаторът M\_ISP-2 да осъществи комуникацията, същият известява за отпадналата свързаност по всеки свой виртуализиран подинтерфейс, показано на фигура 13. Чрез HSRP целевата комуникация автоматично се поема от маршрутизатора M\_ISP-1, преминаващ от статут „Standby“ в режим „Active“ за съответния еднозначно отговарящ подинтерфейс, което е категорично онагледено на фигура 14.

```
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1, changed state to down
%LINK-3-UPDOWN: Interface GigabitEthernet0/0/1.10, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.10, changed state to down
%LINK-3-UPDOWN: Interface GigabitEthernet0/0/1.99, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.99, changed state to down
%LINK-3-UPDOWN: Interface GigabitEthernet0/0/1.192, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.192, changed state to down
%LINK-3-UPDOWN: Interface GigabitEthernet0/0/1.200, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/1.200, changed state to down
```

**Фиг. 13.** Информация за промяна на статута на M\_ISP-2

```
M_ISP-1#
%HSRP-6-STATECHANGE: GigabitEthernet0/0/0.200 Grp 4 state Standby -> Active
```

**Фиг. 14.** Информация за промяна на статута на M\_ISP-1

В този случай трасето на комуникацията се променя (фигура 15), като достъпът до крайна дестинация и необходимата информация се запазва (фигура 16).

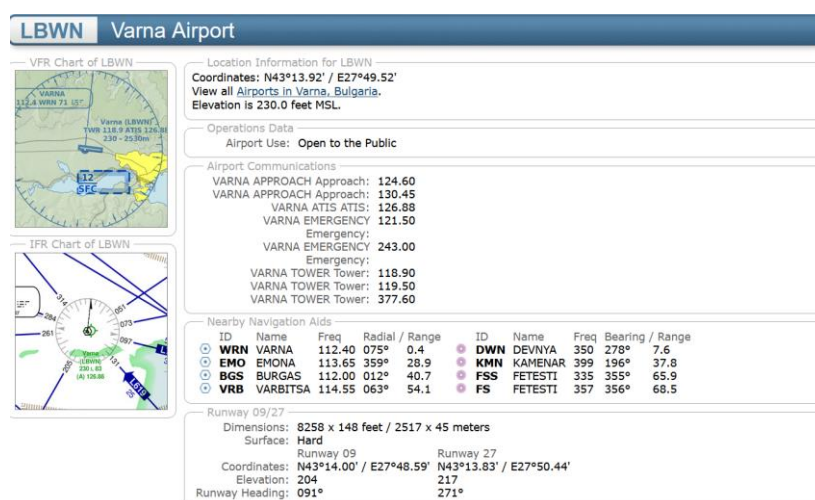
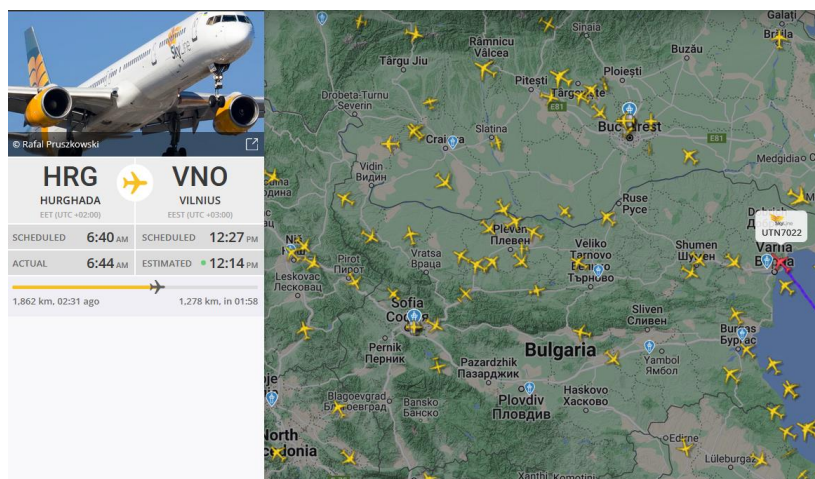
```
C:\>tracert 184.16.60.101

Tracing route to 184.16.60.101 over a maximum of 30 hops:

  0  0 ms    0 ms    0 ms    78.83.151.2
  1  1 ms    0 ms    1 ms    46.10.149.17
  2  1 ms    1 ms    0 ms    184.16.60.101

Trace complete.
```

**Фиг. 15.** Информация за трасето на комуникацията с използване на резервен маршрут



Фиг. 16. Информация за достъп до публичен ресурс в реално време и такъв с константен статут

Аналогични проверки при подобни обстоятелства доведжат до идентичен резултат.

### 3. Заключение

Настоящото емпирично изследване е свързано с реални практики и налага изводите:

1. При съществуващите предизвикателства от обективен или субективен характер пред редица организации, планиращи експлоатация на комуникационното си оборудване за гарантиран достъп до целеви публични ресурси със селектирано съдържание, мрежовото резервиране в граничните точки на свързаност е целесъобразно като категорично изискване.
2. С възможностите за пренос от различни доставчици на публични услуги през разнообразни комуникационни среди, е необходимо налагането на стандартизирани (включително в международен аспект) инструменти, технологични протоколи и организационни процедури. Необходимо е изграждането на инфраструктурите да се базира на проверено оборудване в експлоатационна поддръжка от производителя.
3. Детайлните конфигурации на устройствата, разработени от автора и доказали ефективността си, потенциално позволяват внедряване (със съответно преадресиране) при мрежово изграждане или миграция на организации или учреждения с два Интернет доставчика.
4. Типовата топология, обект на настоящата статия, е база за по-нататъшни разработки по варианти за вътрешна резервираност и балансиране на свързаността по видове трафик.

## Литература

- [1]. European patent Specification, Application number: 04015335.5, Inventors: Darling, Christopher L. Sammamish, Washington 98074 (US); Joy, Joseph M. Redmond, Washington 98052 (US); Shrivastava, Sunita Redmond, Washington 98052 (US), Subbaraman, Chittur Carnation, Washington 98014 (US); Representative: Grünecker, Kinkeldey, Stockmair & Schwanhäusser Anwaltssozietät Leopoldstrasse 4 80802 München (DE). 03.09.2008, Bulletin 2008/36.
- [2]. ISO/IEC/IEEE 8802-1Q:2020/Amd 31:2021, Telecommunications and exchange between information technology systems – Requirements for local and metropolitan area networks – Part 1Q: Bridges and bridged networks – Amendment 31: Stream Reservation Protocol (SRP) enhancements and performance improvements, 2021.
- [3]. ISO/IEC 21823-3:2021, Internet of things (IoT) – Interoperability for IoT systems – Part 3: Semantic interoperability, 2021.
- [4]. ISO/IEC/IEEE 8802-3:2021, Telecommunications and exchange between information technology systems - Requirements for local and metropolitan area networks - Part 3: Standard for Ethernet, 2021.
- [5]. Официален WEB сайт на Международна организация по стандартизация / секция „35.110 Работа в мрежа Включително локални мрежи (LAN), градски мрежи (MAN), широкообхватни мрежи (WAN) и др.“, <https://www.iso.org/ics/35.110/x/>, Последно посетен на 20.03.2024 г.
- [6]. D. Awduche, J. Malcolm, J. Agogbua, M. O’Dell, and J. McManus, „Requirements for Traffic Engineering Over MPLS,” Internet Request for Comments, RFC Editor, RFC 2702.
- [7]. G. Malkin, Xylogics, Inc., T. LaQuey Parker, UTexas, Editors, January 1993, Request for Comments: 1392, Internet Users' Glossary, Network Working Group, Internet Engineering Task Force (IETF).
- [8]. C.M. Heard, Ed., Consultant, September 2003, Request for Comments: 3637, Definitions of Managed Objects for the Ethernet WAN Interface Sublayer, Copyright (C), Network Working Group, Internet Engineering Task Force (IETF), The Internet Society.
- [9]. Charles Hornig, Symbolics Cambridge Research Center, April 1984, Request for Comments: 894, A Standard for the Transmission of IP Datagrams over Ethernet Networks, Network Working Group, Internet Engineering Task Force (IETF), Last Updated 2020-07-29.
- [10]. Cisco 4000 Family Integrated Services Router, Data Sheet, © 2021 Cisco and/or its affiliates, Printed in USA C78-732542-24 08/2; [https://www.cisco.com/c/en/us/products/collateral/routers/4000-series-integrated-services-routers-isr/data\\_sheet-c78-732542.html](https://www.cisco.com/c/en/us/products/collateral/routers/4000-series-integrated-services-routers-isr/data_sheet-c78-732542.html) . Последно посетен на 30.03.2024 г.
- [11]. End-of-Sale and End-of-Life Announcement for the Cisco ISR4200, ISR4300 and select ISR4400 Series Platform, © 2023 Cisco and/or its affiliates. All rights reserved, <https://www.cisco.com/c/en/us/products/collateral/routers/4000-series-integrated-services-routers-isr/select-isr4k-series-platform-eol.html>. Последно посетен на 30.03.2024 г.
- [12]. Cisco Catalyst 3650 Series Switches, Data Sheet, © 2021 Cisco and/or its affiliates, Printed in USA C78-729449-22 04/18, [https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-3650-series-switches/data\\_sheet-c78-729449.html](https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-3650-series-switches/data_sheet-c78-729449.html). Последно посетен на 01.04.2024 г.
- [13]. End-of-Sale and End-of-Life Announcement for the Cisco Catalyst 3650, © 2022 Cisco and/or its affiliates. All rights reserved. C51-744426-02 11/22, <https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-3650-series-switches/eos-eol-notice-c51-744426.html>. Последно посетен на 02.04.2024 г.

- [14]. Y. Rekhter, Ed., T. Li, Ed., S. Hares, Ed., Request for Comments: 4271, Obsoletes:1771, Copyright (C) The Internet Society (2006), A Border Gateway Protocol 4 (BGP-4), Network Working Group, Internet Engineering Task Force (IETF).
- [15]. Understand the Hot Standby Router Protocol Features and Functionality, Document ID: 9234, Revision 1.0 02-Dec-2013 Initial Release, Revision 3.0 05-Aug-2022 Initial Release, Revision 4.0, 13-Sep-2023 Recertification, Updated: September 13, 2023, [https://www.cisco.com/c/en/us/support/docs/ip/hot-standby-router-protocol-hsrp/9234-hsrpguidetoc.html?referring\\_site=bodynav](https://www.cisco.com/c/en/us/support/docs/ip/hot-standby-router-protocol-hsrp/9234-hsrpguidetoc.html?referring_site=bodynav). Последно посетен на 04.04.2024 г.
- [16]. Review Hot Standby Router Protocol (HSRP), FAQ, Document ID: 9281, Revision 1.0 21-Nov-2001 Initial Release, Updated: June 26, 2023, <https://www.cisco.com/c/en/us/support/docs/ip/hot-standby-router-protocol-hsrp/9281-3.html>, Последно посетен на 04.04.2024 г.
- [17]. E. Bell - 3Com Corp., A. Smith - Extreme Networks, P. Langille - Newbridge Networks, A. Rijhsinghani - Cabletron Systems, K. McCloghrie - cisco Systems. Request for Comments: 2674, Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions, Network Working Group, Internet Engineering Task Force (IETF), August 1999, <https://www.ietf.org/proceedings/61/RFCs/rfc2674.txt>, Последно посетен на 08.04.2024 г.

**За контакти:**

инж. Георги С. Николов  
докторант в катедра „Комуникационна техника и технологии“  
Технически университет - Варна  
e-mail: [gnikolov@tu-varna.bg](mailto:gnikolov@tu-varna.bg)



# МЕТОДИКА ЗА ОБУЧЕНИЕ НА АСЕМБЛЕРЕН ЕЗИК

Жейно Ив. Жейнов, Десислава Д. Ангелова

**Резюме:** Разглежда се въведената наскоро методика за обучение на студентите на езика за програмиране Асемблер по дисциплината „Микропроцесорна техника”, изучавана от студентите от втори курс на специалностите „КСТ” и „СИТ” при ТУ-Варна. Прави се кратък преглед на изучаваните микропроцесори и техния асемблерен език, както и на използвания инструментариум при обучението. Разглеждат се основните похвати, използвани при програмирането, структурите от данни, типовите решавани задачи, начините за представяне на алгоритмите и как се получава машинният код от текста на асемблерната програма. Описва се хардуерното и софтуерното осигуряване, нужно за програмиране на Асемблер и за връзката му с език от високо ниво.

**Ключови думи:** програмиране на Асемблер, алгоритми, типове решавани задачи, връзка с език от високо ниво, хардуерно и софтуерно осигуряване на обучението.

## Assembly language teaching methodology

Zhejno I. Zhejnov, Desislava D. Angelova

**Abstract:** The recently introduced methodology for training the students in the programming language Assembler in the discipline "Microprocessors", studied by the students of the second year of the specialties "KST" and "SIT" at TU-Varna, is examined. A brief overview of the studied microprocessors and their Assembly language is given, as well as the tools used in the training. The basic concepts used in programming, data structures, typical problems solved, ways of representing algorithms and how to obtain machine code from the text of the Assembly program are considered. Describes the hardware and software required to program Assembler and to interface it with a high-level language.

**Keywords:** Assembler programming, algorithms, problem types, high-level language connection, hardware and software support for training.

## 1. Увод

Дисциплината “Микропроцесорна техника” се изучава от студентите втори курс от специалностите „КСТ” и „СИТ” на ТУ-Варна.

Преди няколко години в нея бяха изучавани само 16-битовите CISC микропроцесори на Intel, използвани в персоналните компютри (PC) в България. Развитието на интегралната схемотехника и създаването на други PC, изградени на основата на нови микропроцесори, наложи обновяване на учебния материал. Новите проекти днес се управляват чрез микропроцесорни системи, разработени на основата на CISC и RISC микропроцесори.

Дисциплината е посветена на популярните 32-битови микропроцесори Intel 80x86 и Arm Cortex A-8. Разглежда се архитектурата на двата вида популярни CISC и RISC микропроцесори и изграждането на микропроцесорни системи с тях. Описват се основните устройства в структурата им, системните шини, входно-изходната памет. Изучават се системата от машинни команди, методите за адресиране, системата за прекъсване и възможностите за обмен на информация с външната среда. Изучава се асемблерният език за програмиране и работата със съответните програмни среди за създаване и настройка на асемблерни програми. Разглеждат се особеностите на 64-битовите микропроцесори.



## 2. Изложение

Основна част от обучението е посветено на изучаването на особеностите при програмирането на Асемблерен език за двата типа микропроцесори: IA-32 и Arm Cortex A-8. Разглеждат се последователно програмните модели, типовете и структурите на използваните данни, директивите за описание на данните. Изучават се мнемоничните кодове на командите за прехвърляне на данните, за аритметични операции, за управление на преходите, за цикли, за битови операции, за извикване на подпрограми. Специално внимание се отделя на предаването на параметрите между главната програма и подпрограмите и на работата със стека. Изучава се структурата на многомодулните програми и връзката между модулите. Разглеждат се операциите за вход-изход на данни на ниво порт и ниво потребителска програма. Показва се представянето на данните в паметта на микропроцесорната система в двоичен и VCD код. Разглеждат се извикването на командите за копроцесорите за Intel и Arm. Описва се разработката на асемблерни програми. Дават се начините за създаване и тестване и процедурите за трансляция, свързване, тестване и настройка при двата разглеждани процесора. Обучението е обезпечено с 2 учебни помагала - ръководства за лабораторни упражнения по Микропроцесорна техника и за програмиране на Асемблер, съответно за Intel и Arm процесори [2], [3].

Изучаването на програмиране на Асемблер (в сравнение с програмиране на език от високо ниво) дава следните предимства:

1. изяснява изпълнението на инструкциите;
2. показва как данните са представени в паметта;
3. показва взаимодействието на програмата с операционната система, процесора и I/O система;
4. изяснява как една програма осъществява достъп до външни устройства.

Програмистите на Асемблер стават по-добри програмисти на език от високо ниво (HLL). Те получават по-добра представа за целевия език, на който HLL се превежда.

Асемблерният език е език за програмиране, който е на крачка от машинния език. Обикновено всяка инструкция на асемблерния език се превежда в една машинна инструкция от Асемблера. Асемблерният език зависи от хардуера. Всеки тип процесор има различен асемблерен език. Инструкциите на асемблерния език могат да се позовават на конкретни регистри в процесора, да включват всички кодове на операции на процесора и да отразяват дължината на битовете на различните регистри на процесора и операндите на машинния език. Затова програмистът на Асемблер трябва да разбира архитектурата на компютъра.

Програмистите рядко използват Асемблер за приложения или дори за системни програми. HLL осигуряват изразителна сила и стегнатост, които значително улесняват задачите на програмиста. Недостатъците на използването на Асемблер вместо HLL са [1]:

1. Време за разработка. Писането на код на Асемблер отнема много повече време, отколкото писане на език от високо ниво.
2. Надеждност и сигурност. Лесно е да се правят грешки в асемблерния код. Асемблерът не проверява дали се спазват конвенциите за извикване и за запис на регистър. Никой не проверява вместо вас дали броят на PUSH и POP инструкциите е еднакъв във всички възможни разклонения и пътища. Има много възможности за скрити грешки в асемблиращия код. Това влияе на надеждността и сигурността на проекта, освен ако имате много систематичен подход към тестването и проверката.
3. Отстраняване на грешки и проверка. Кодът на асемблирането е по-труден за отстраняване на грешки и проверка, тъй като има повече възможности за грешки, отколкото в кода на високо ниво.

4. Поддръжка. Асемблерният код е по-труден за модифициране и поддръжка, защото езикът позволява неструктуриран код и трикове, трудни за разбиране от другите. Необходима е изчерпателна документация и последователен стил на програмиране.

5. Преносимост. Кодът за асемблиране е специфичен за платформата. Пренасяне към друга платформа е трудно.

6. Системният код може да използва вътрешни функции вместо асемблиране. Най-добрите модерни C++ компилатори имат присъщи функции за достъп до системните контролни регистри и други системни инструкции. Кодът за асемблиране вече не е необходим за устройството за драйвери и друг системен код, когато са налични вътрешни функции.

7. Кодът на приложението може да използва вътрешни функции или векторни класове вместо асемблиране. Най-добрите съвременни C++ компилатори имат вградени функции за векторни операции и други специални инструкции, които преди това изискваха програмиране на Асемблер.

8. Компилаторите са подобрени много през последните години. Необходими са много експертизи и опит, за да се оптимизира по-добре от най-добрия C++ компилатор.

Използването на Асемблер дава някои предимства:

1. Отстраняване на грешки и проверка. Разглеждането на генерирания от компилатора асемблен код в програма за тестване и настройка (Debugger) е полезно за намиране на грешки и за проверка колко добре компилаторът оптимизира определена част от кода.

2. Изработване на компилатори. Необходимо е за разбирането на техниките за кодиране на асемблиране за създаване на компилатори, програми за отстраняване на грешки и други инструменти за разработка.

3. Вградени системи. Малките вградени системи имат по-малко ресурси от компютрите и мейнфрейми. Програмирането на Асемблер може да е необходимо за оптимизиране на код за скорост или размер в малки вградени системи.

4. Хардуерни драйвери и системен код. Достъпът до хардуер, системни контролни регистри и т.н. понякога може да бъде труден или невъзможен с код от високо ниво.

5. Достъп до инструкции, които не са достъпни от език на високо ниво. Някои инструкции за асемблиране нямат езиков еквивалент на високо ниво.

6. Самомодифициращ се код. Самомодифицирането на код обикновено пречи на ефективното кеширане на код. Може да е изгодно за включване на малък компилатор в математически програми, дефинирани от потребителя, където една функция трябва да се изчислява много пъти.

7. Оптимизиращ код за размер. Мястото за съхранение и паметта са много евтини ресурси в днешно време и не си струва използването на Асемблер за намаляване на размера на кода. Размерът на кеша обаче все още е критичен ресурс и може в някои случаи да бъде полезен за оптимизиране на критична част от кода по размера на кеша.

8. Оптимизиращ код за скорост. Съвременните C++ компилатори обикновено оптимизират кода доста добре в повечето случаи. Все още има случаи, в които компилаторите работят лошо и където може да се постигне драстично увеличение на скоростта чрез внимателно асемблерно програмиране.

9. Библиотеки с функции. Общата полза от оптимизирането на кода е по-висока при създаване на библиотеки с функции, които се използват от много програмисти.

10. Създаване на функционални библиотеки, съвместими с множество компилатори и работа системи. Възможно е да се правят библиотечни функции с множество записи, които да са съвместими с различни компилатори и различни операционни системи. Това изисква програмиране на Асемблер.

Асемблерът е програмата, която превежда асемблерния език в машинен код.

Асемблерният език е символно представяне на машинния език на конкретен процесор, допълнено от допълнителни типове изрази, които улесняват писането на програма и предоставят инструкции на Асемблера.

Машинен език или машинен код е двоичното представяне на компютърна програма, която всъщност се чете и интерпретира от компютъра. Изпълнимият код е машинният код, генериран от езиков процесор за изходен код, като например Асемблер или компилатор. Това е софтуер във форма, която може да се стартира на компютъра.

Обектният код е машинно езиково представяне на програмния изходен код. Обектният код се създава от компилатор или Асемблер и след това се превръща в изпълним код от свързващия редактор (Linker). Зареждащата програма е програмна процедура, която копира изпълнима програма в паметта за изпълнение.

При разработката на Асемблерна програма студентите правят следното:

- Съставят блокова схема на алгоритъма. Функционалните блокове в блоковата схема описват инструкциите на използвания процесор;

- От блоковата схема се разписват мнемоничните кодове на програмата съгласно списъка с инструкции за съответния вид процесор, приложен в ръководството за лабораторни упражнения и материалите за съставяне на сорса;

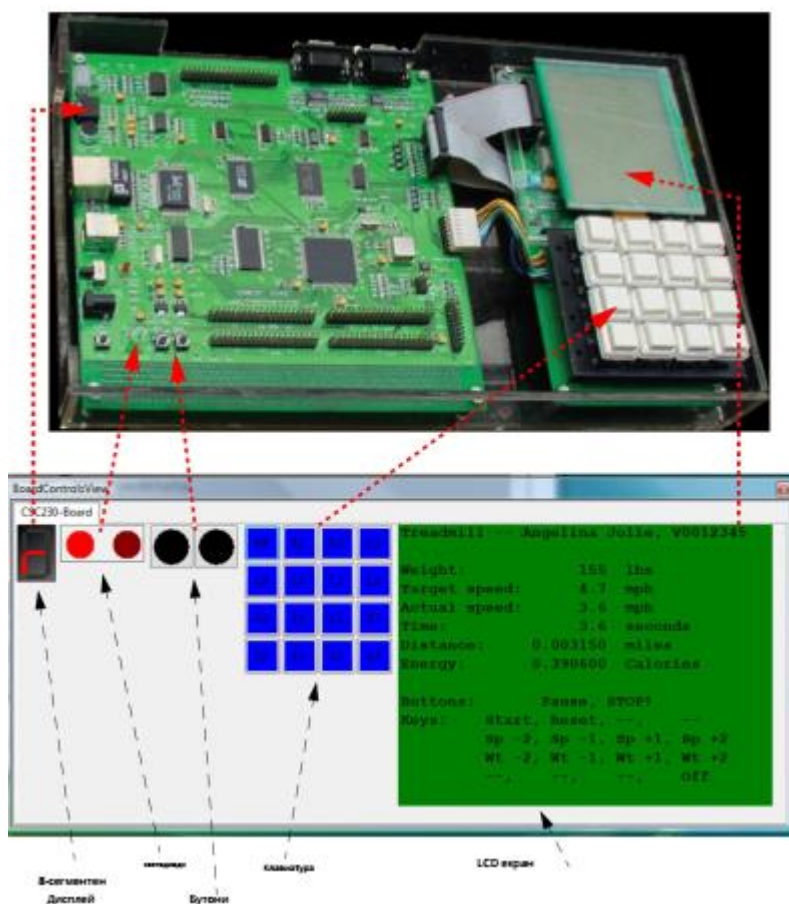
- Въвежда се получената Асемблерна програма с текстов редактор в РС и се записва на диска му като текстов файл със съответно разширение. В зависимост от използвания микропроцесор, за който е съставена програмата на Асемблер, се действа различно:

При IA-32 процесори се ползва инсталираният на РС безплатен софтуер Microsoft Masm32 [4]. Той представлява интегрирана програмна среда (IDE), която включва текстов редактор, транслатор и свързващ редактор. Средата се стартира чрез извикване на изпълнимия файл от пакета qeditor.exe. Асемблерната програма се въвежда и записва на диска като текстов файл с разширение .asm чрез вградения текстов редактор на средата. С натискането на бутон Project\Assemble ASM file от менюто на IDE средата се транслира полученият файл до обектен с разширение .obj, ако няма грешки в сорса. После с натискане на бутон Project\ Assemble & Link или Project\Build All на диска се създава файл с изпълним код (с разширение .exe). Ако транслаторът намери грешки при транслирането и свързването на сорса, се извеждат съобщения за вида на грешката. Програмистът трябва да коригира сорса и пак да транслира и свързва Асемблерната програма, докато бъдат отстранени всички намерени грешки.

После се извършва проверката/тестването/ на получения машинен код. Файлът с машинната програма се въвежда в паметта с помощта на отделно приложение /Olly Dbg ver. 2.01/. Тази програма за тестване и настройка е безплатно Windows приложение, което има всички възможности на класическата програма за тестване. Тя позволява разглеждане и промяна на паметта с машинния код и данните на Асемблерната програма, както и състоянието на регистрите и стека, дизасемблиране на кода, включително и на FPU инструкциите, постъпково изпълнение на програмата, поставяне на точки на прекъсване, изпълнение на програмата от началото до края, както и много други полезни функции. Ако програмата прави онова, за което е била написана, работата на програмиста приключва. Иначе той трябва да коригира съобразно грешката блоковата схема, сорса на програмата, да го запише на диска на РС и отново да я транслира и тества работата на машинния код. Тези корекции на кода и проверка на действието на машинната програма чрез приложението Debugger се правят дотогава, докато се получи правилно работеща Асемблерна програма.

При Arm процесорите студентите трябва да програмират същата задача на Асемблер, която е решавана при IA-32 процесора. Програмата ще се изпълнява на Arm микропроцесорна система, реализирана с други компоненти. Проверката на правилността на написаната програма се осъществява, като се ползва програмен симулатор на микропроцесорна система с Arm микропроцесор (ARMSim#191). Това безплатно програмно

Windows приложение, описано в [5], се ползва за симулация на описаната в [6] ARM микропроцесорна система ARM7TDMI с включен входно-изходен модул Embest – фигура 1.



**Фиг.1.** Развойна платка Embest и симулирането ѝ

Микропроцесорната система ползва развойна платка Embest със 7-сегментен светодиоден индикатор, 2 бутона, 2 светодиода, 4x4 бутонна клавиатура и течнокристален LCD дисплей.

Симулаторът позволява проверка и тестване на написаните ARM асемблерни програми, като за целта в него се въвежда Асемблерният сорс - текстов файл с разширение .s, създаден с обикновен текстов редактор - например с Notepad или с текстовия редактор на Microsoft MASM32 средата. Този симулатор е ползван за обучение по програмиране на Асемблер в университета Виктория в Канада. Той предлага редица предимства:

1. Автоматизирано тестване на ARM програми. Те се зареждат като файл с Асемблерен сорс-код. Потребителят не транслира програмата и не зарежда машинен код в паметта. Симулаторът изписва на реда със сгрешените оператори на програмата вида на грешката. Позволява удобно тестване и настройка на Асемблерни ARM програми. Симулаторът дава възможност за разглеждане и промяна на паметта с машинния код и данните на Асемблерната програма. Извежда състоянието на регистрите и стека. Позволява поставяне на точки на прекъсване, изпълнение на програмата по стъпки и от началото до края.

2. Предоставя предварително дефинирани функции като plug-ins, извиквани с swi инструкция. Те извеждат състоянието на натиснатите бутони, индицират зададени чрез регистрите седемсегментни кодове, изписват дефинирани стрингове в програмата като надписи върху LCD екрана или на работното поле на симулатора /работещ като Debugger/.

3. Дава възможност за работа у дома, тъй като не е необходим развоен макет с ARM микропроцесорна система и развоен софтуер, което позволява лесно усвояване на основните ARM инструкции от Асемблера без допълнителни разходи.

Тази версия на симулатора обаче не позволява проверка на програмирането на включения в архитектурата на Arm микропроцесора FPU, както и не могат да се изпълнят програми, съдържащи някои специални инструкции. За професионална употреба се изисква управление с асемблерни програми на работата на развоен микропроцесорен макет. За тази цел може да се използва реална микропроцесорна система с включени допълнителни I/O елементи. Множество фирми предлагат свои развойни платки с включено инструментално програмно осигуряване. Използването на такъв макет е удобно и за изучаване на Асемблер и за връзката му с други HLL езици за програмиране.

В този случай може да се използва прост ARM PC, като например предлагания на нашия пазар евтин Raspberry Pi. Транслирането, свързването и настройката на асемблерната програма в този случай би могло да се направи под управлението на операционната система с вграден Асемблер, като например GNU Асемблер под Linux [7].

Има много добра документация за извикването на C функции в Linux среда. С използването на такъв кит става възможно изпълнението на всички асемблерни инструкции, които не могат да бъдат пуснати на симулатора без сериозни софтуерни корекции – например инструкции за FPU и NEON модулите и за управление на I/O крачета на ARM [8].

### Заклучение

Описаната методика за обучение на Асемблер позволява лесно и евтино обучение на Асемблер за два различни съвременни микропроцесора. Студентите усвояват особеностите на асемблерния език и мястото му в съвременните разработки.

### Литература

- [1]. Stalling, W.. Computer organization and architecture. Designing for Performance. Pearson Education, Inc., USA, 2016, pp. 864.
- [2]. Тянев, Д., Ж. Жейнов Микропроцесорна техника и програмиране на Асемблер. Университетско изд. при ТУ-Варна, 2014, pp. 280.
- [3]. Георгиев, Л., П. Генчев. Микропроцесорна техника. Програмиране на Асемблер за 32-битови RISC-процесори „ARM”. Университетско изд. при ТУ-Варна, 2022, pp. 140.
- [4]. Bray, B. V.. The Intel microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro processor, Pentium II, Pentium III, Pentium 4, and Core2 with 64-Bit extensions: architecture, programming, and interfacing. London, Pearson Education Ltd., 2009, pp. 944.
- [5]. Horspool, R. N., W. D. Lyons, M. Serra. ARMSim# - a customizable simulator for exploring the ARM architecture. B: FECS'09, Canada, University of Victoria, 2009, pp 5.
- [6]. Horspool, R. N., W. D. Lyons, M. Serra. ARMSim# User Guide. Canada, University of Victoria, Department of Computer Science, 2010, pp 47.
- [7]. Pyeatt, L. D.. Modern Assembly Language. Programming with the ARM Processor. USA, Elsevier Inc., 2016, pp. 476.
- [8]. Smith, S.. Raspberry Pi Assembly Language Programming. ARM Processor Coding. USA, Springer Science, 2019, pp. 323.

#### За контакти:

доцент д-р Жейно Ив. Жейнов  
катедра „Компютърни науки и технологии”  
Технически Университет - Варна  
E-mail: zh\_viv@tu-varna.bg

# ПРИЛОЖЕНИЕ НА СИМЕТРИЧЕН КРИПТОГРАФСКИ АЛГОРИТЪМ AES НА 8 БИТОВ PIC МИКРОКОНТРОЛЕР

Пламен Й. Стоянов

**Резюме:** Криптографският алгоритъм AES (Advanced Encryption Standart) е индустриален стандарт за криптиране и декриптиране. Той е широко използван в различни криптографски приложения като безжични сензорни мрежи, Smart карти, RFID и др. Този алгоритъм притежава висока ефективност при различни платформи от 8 битови до 64 битови процесори. В статията се предлагат ефективни решения за повишаване на бързодействието при криптиране и декриптиране на 128 битов блок от данни. Повишаване на бързодействието води до намаляване на консумацията на енергия и увеличаване на сигурността. Предложените решения са сравнени в табличен вид с публикувани изследвания за ускорено изпълнение, като основен показател е броят цикли за изпълнение.

**Ключови думи:** криптографски алгоритъм AES, софтуерно приложение, 8 битов микроконтролер, сравнителен анализ.

## Implementation of symmetric cryptographic algorithm AES of a 8-bit PIC Microcontroller

Plamen I. Stoianov

**Abstract:** Cryptographic algorithm AES (Advanced Encryption Standart) is the industry standart for encryption and decryption. AES is most widely used in cryptographic applications typical for wireless sensor nodes, Smart cart, RFID etc. It is extremely efficient on many different platforms ranging to an 8-bit to 64-bit processors. This paper presents effective solutions to increase the speed of encryption and decryption of a 128-bit block of data. Increase in speed leads to a decrease in energy consumption and increase in security. The proposed solutions are compared with published research on accelerated execution. The main indicator is the number of cycles for execution.

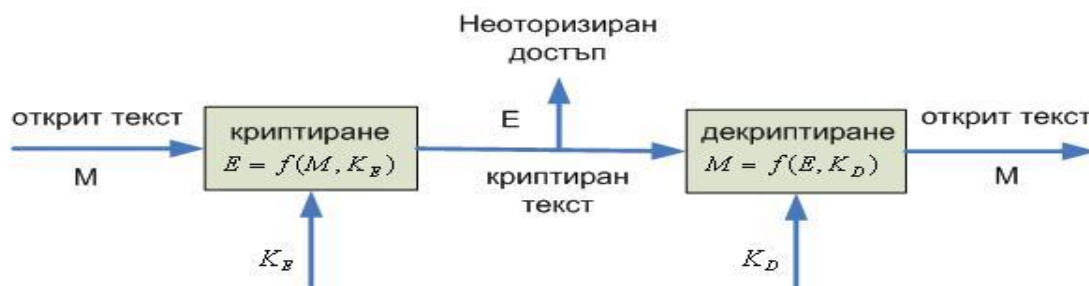
**Keywords:** cryptographic algorithm AES, Software implementation, 8-bit microcontroller, comparative analysis.

### 1. Увод

Думата криптография има гръцки произход и означава тайно писане (*κρυπτός, криптос* – „скрит“, и *γράφω, графо* – „пиша“). Задача на криптографията е преобразуване на открит текст (plaintext) в шифриран текст (ciphertext) чрез използване на криптографски алгоритъм и еднозначно възстановяване на оригиналния текст. Процесът на преобразуване в шифриран текст се нарича шифриране или криптиране (encryption), докато обратният процес на възстановяване е дешифриране или декриптиране (decryption). Задължително изискване е възможността еднозначно и пълно възстановяване на открития текст от шифрирания текст чрез определени логически операции. За преобразуване на данните се използва криптографски алгоритъм, основаващ се на определена математическа функция за шифриране и дешифриране [1].

Съвременните изисквания са довели до появата на нетрадиционни задачи за защита на данните. Типичен пример е автентикация в случаите, в които двете страни, обменящи информация, нямат доверие една на друга. Решението на този проблем е използване на криптографски алгоритми с два различни ключа – един за криптиране и втори за декриптиране. Началото е поставено от американските учени Diffie и Hellman през 1976г.

Процесът по криптиране и декриптиране на данни е представен на фигура 1 в блоков вид. Входните данни  $M$ , с помощта на ключ за криптиране  $K_E$ , се преобразуват в криптиран вид  $E$ , чрез използване на определена математическа функция  $f$ . В приемната страна криптираната информация се възстановява, тъй като  $M$  е функция на криптирания текст  $E$  и ключ за декриптиране  $K_D$ . Има два основни типа криптографски алгоритми – симетрични и асиметрични. Това разделяне е според отношението им към ключовете за криптиране и декриптиране. Ако тези ключове  $K_E$  и  $K_D$  съвпадат, алгоритъмът е симетричен или още алгоритъм със секретен ключ. В този случай е необходим секретен канал за обмен на сесийни ключове.



Фиг. 1. Обща блок-схема на криптографска система

## 2. Изложение

Най-широко използваният симетричен алгоритъм е DES (Data Encryption Standard) разработен от IBM. DES е от типа шифри на Фейстел (Horst Feistel- криптограф работил в IBM), при които откритият текст се разделя на два блока, които се обработват отделно. Този алгоритъм преобразува 64-битов открит текст в 64-битов шифротекст. Ключът е 64-битов, от които 56 бита се използват директно в алгоритъма. За увеличаване разрядността на ключа и повишаване на устойчивостта, към методите за криптоанализ са разработени различни варианти. Използват се различни подключове в отделните етапи или допълнителен ключ (при DESX е 64 бита) за промяна на входните и изходни данни. За увеличаване разрядността на ключа най-често се използва TDES (Triple-DES) известен като 3DES. Извършва се последователно криптиране-декриптиране-криптиране на данните с два или три независими ключа. Поради ниската размерност на ключа и проведените успешни крипто атаки, NIST (National Institute of Standards and Technology) стартира процедура по замяна на този стандарт. През 2000г. е избран алгоритъм Rijndael, и през следващата година е стандартизиран като AES (Advanced Encryption Standard) чрез FIPS PUB 197 [2]. Досега няма данни за извършени успешни атаки срещу AES с изключение на използването на съкратени варианти, т.е. по-малко на брой етапи от предвидените в стандарта 10, 12 и 14 [3, 4]. Целта на всички софтуерни изпълнения е ускоряване на процеса на криптиране и декриптиране при възможно по-малка по обем програма. За реализация на AES е избран микроконтролер PIC18 на фирмата MICROCHIP. Той притежава три отделни индексни регистъра с възможност за автоматичното им увеличаване и намаляване, както и регистър TBLPTR за работа с таблици от данни. Това дава възможност за лесна и бърза обработка на няколко масива.

При алгоритъм AES се извършват действия с полиноми в поле на Галоа  $GF(2^8)$ . Данните се представят като полиноми със степен по-малка от 7 и коефициенти в полето  $GF(2)$ . Междинните резултати при обработката се наричат състояние (State) и се представят като квадратна матрица от 4 реда и 4 колони. Всеки байт от състоянието се бележи със  $s_{r,c}$ , където  $r$  е редът и  $c$  е колоната от 0 до 3. Данните се обработват в 10, 12 или 14 еднотипни

етапа -  $N_r$ , като броят им зависи от разрядността на ключа – 128, 192 или 256 бита. В зависимост от ключа, стандартът определя три варианта: AES-128, AES-192 и AES-256. Всеки етап се състои от 4 различни обработки на състоянието - SubBytes, ShiftRows, MixColumns, AddRoundKey, и допълнителна за получаване на подключовете за всеки етап – ExpandKey. Последователността на изпълнение на алгоритъма е представена като псевдо код:

---

### Алгоритъм 1 - AES algorithm

---

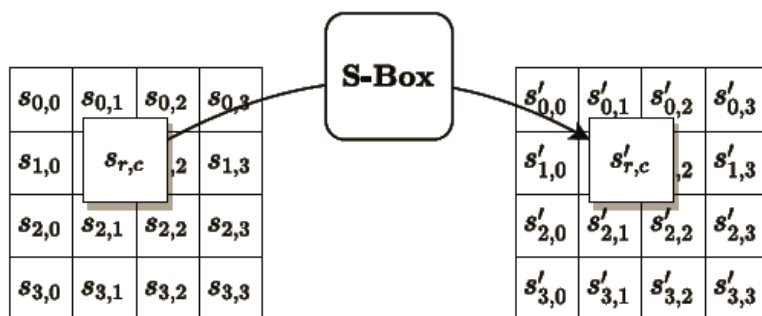
```

procedure AES (state, key)
  1: state ← AddRoundKey(state,key)
  2: key ← ExpandKey(key)
  3: for i = 1 to  $N_r - 1$  do
  4:   state ← SubBytes(state)
  5:   state ← ShiftRows(state)
  6:   state ← MixColumns(state)
  7:   state ← AddRoundKey(state,key)
  8:   state ← ExpandKey(key)
  9: end for
  10: state ← SubBytes(state)
  11: state ← ShiftRows(state)
  12: state ← AddRoundKey(state,key)
  13: return state
end procedure

```

---

Алгоритъмът може да се раздели на 3 етапа. В първия етап се копират входните данни в областта на състоянието (State) и се сумират по mod2 с началния ключ – AddRoundKey (линия 1). Изчислява се ключът за следващата обработка ExpandKey (линия 2). При втория етап се извършват всичките 4 обработки на state и разширение на ключа за следващия етап (линии 4 до 8). Вторият етап се извършва в цикъл от 9, 11 или 13 пъти в зависимост от размерността на ключа. Финалният етап (линии 10, 11, 12) се изпълнява извън цикъла, тъй като не е необходим ключ за следващия етап и не се изпълнява MixColumns. В отделните трансформации и получаване на подключове се изпълняват следните действия:



Фиг. 2. Изпълнение на преобразуване SubBytes

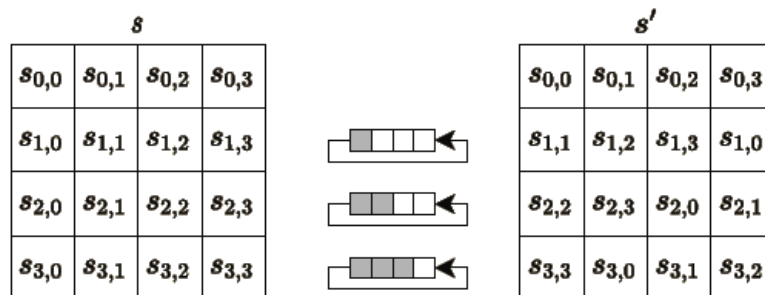
- **SubBytes** – за получаване на изходната стойност се извършват две отделни изчисления: мултипликативна инверсия и свързано преобразуване. Ако  $a(x)$  е входният полином, то мултипликативната инверсия  $a(x)^{-1}$  се намира чрез  $a(x).a(x)^{-1} \equiv 1 \pmod{m(x)}$ . Използва се неприводим полином  $m(x) = x^8 + x^4 + x^3 + x + 1$ . При размерност на полето от



(2<sup>8</sup>) елемента, е удачно предварително изчисление на стойностите и запис в паметта като таблица с размерност 256 байта. Това води до по-висока скорост на изпълнение и по-малък обем на използваната програмна памет. Фигура 2 илюстрира замяната на байт от матрицата.

При изпълнение на свързано преобразуване всеки бит на изходната стойност се определя от сума по mod2 между определени битове на входната стойност и зададена константа. При модулна инверсия се използва таблица, следователно нейните стойности може да се преизчислят според условията на свързаното преобразуване. По този начин двете операции се изпълняват само с едно индексирание на таблица. За процеса на декриптиране е необходима допълнителна таблица. За извличане от таблица е необходим предварителен запис в регистър TBLPTR, поради което за замяна на един байт са нужни 6 цикъла на микроконтролера. За един ред или стълб са необходими 24 цикъла, а за цялата матрица - 96 цикъла.

- **ShiftRows** – извършва се ротация на думи в отделните редове на матрицата. Нулевият ред за всеки етап остава без промяна, докато 1-ви, 2-ри и 3-ти ред от матрицата се ротира наляво, съответно на 1, 2 и 3 байта. Фигура 3 илюстрира ротацията на редовете на матрицата.



Фиг. 3. Изпълнение на преобразуване ShiftRows

Преобразуванията SubBytes и ShiftRows може да се обединят, при което се съкращават инструкциите за повторно четене и запис. В резултат общото време за изпълнение изисква 96 цикъла, колкото е при SubBytes.

- **MixColumns** – всеки стълб от матрицата се представя като 4-членен полином  $s(x)$  с коефициенти, които са елементи на полето GF (2<sup>8</sup>). Новата стойност  $s'(x)$  се получава след умножаване на  $s(x)$  със зададен в стандарта полином:

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\} \quad (1)$$

Произведението се редуцира по модул  $(x^4 + 1)$ , т.е:

$$s'(x) = a(x).s(x) \text{ mod } (x^4 + 1) \quad \dots \quad (2)$$

В процеса на декриптиране (InvMixColumns) се използва инверсният полином:

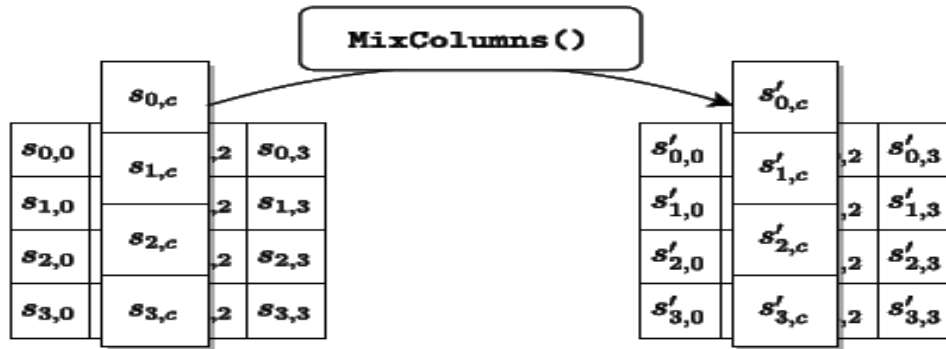
$$a^{-1}(x) = \{0B\}x^3 + \{0D\}x^2 + \{09\}x + \{0E\} \quad (3)$$

получен чрез  $a^{-1}(x).a(x) \equiv 1 \text{ mod } (x^4 + 1)$ .

Входното състояние (State), представено като матрица 4x4, се умножава по фиксирана матрица, представена чрез думата  $[a_0, a_1, a_2, a_3] = [\{02\}, \{01\}, \{01\}, \{03\}]$ . След умножението, новите стойности на съответната колона са:

$$\begin{aligned} s'_{0,c} &= 2.s_{0,c} \oplus 3.s_{1,c} \oplus s_{2,c} \oplus s_{3,c} \\ s'_{1,c} &= s_{0,c} \oplus 2.s_{1,c} \oplus 3.s_{2,c} \oplus s_{3,c} \\ s'_{2,c} &= s_{0,c} \oplus s_{1,c} \oplus 2.s_{2,c} \oplus 3.s_{3,c} \\ s'_{3,c} &= 3.s_{0,c} \oplus s_{1,c} \oplus s_{2,c} \oplus 2.s_{3,c} \end{aligned} \quad (4)$$

Фигура 4 илюстрира преобразуване на колоните на матрицата с входните данни чрез използване на зададения полином в стандарта.



Фиг. 4. Изпълнение на преобразуване MixColumns

Коефициентите на използвания полином се различават само с 1, поради което може да се използва зависимостта  $:c.s_i \oplus (c+1).s_j = c.(s_i \oplus s_j) \oplus s_j$ . Ако предварително се изчисли сумата  $Temp = s_{0,c} \oplus s_{1,c} \oplus s_{2,c} \oplus s_{3,c}$ , то стойността на първия байт от колоната може да се представи чрез  $s'_{0,c} = (s_{0,c} \oplus s_{1,c}).2 \oplus s_{0,c} \oplus Temp$ . Операциите по умножение се намаляват, при запазване броя на сумиранията по mod2. За намиране на четирите байта от колоната са необходими 32 цикъла, следователно за всички 16 байта на матрицата са нужни 128 цикъла.

- **AddRoundKey** – всеки байт от матрицата се сумира по mod2 със съответния байт от текущия подключ, за получаване на новото състояние. Изисква се прочитане и сумиране по mod2, което отнема 32 цикъла за всичките 16 байта. В тялото на цикъла изпълнението на AddRoundKey е непосредствено след MixColumns и следователно двете трансформации може да се обединят. След получаване на новата стойност от смесване на колоните, тя се сумира със съответния байт от ключа и се записва в матрицата като ново състояние. В резултат на обединението на двете трансформации се съкращават инструкциите за четене, при което общото време за изпълнение от  $128 + 32$  цикъла, се намалява на  $128 + 16 = 144$  цикъла.

- **ExpandKey** – за всеки отделен етап се използва различен ключ, получен от основния чрез преобразувания. Както и при данните, областта на ключа се представя като матрица от 4 реда и 4 стълба. След всяко изпълнение на AddRoundKey (с изключение на последния етап), четвъртият ред се ротира наляво на един байт и се извършва SubBytes по същата таблица, както и за данните. Първият байт се сумира с константа, която е различна за отделните етапи. Следва сумиране по mod2 на четвърти към първи, първи към втори, втори към трети, и трети към четвърти ред. По този начин се получават подключовете за всеки следващ етап. Необходимото време за изпълнение на всяка стъпка е 53 цикъла. Те могат да се намалят, ако замяната на байтове се извършва от таблица в данновата памет, вместо от програмната памет. Друг начин за ускоряване е чрез изнасяне на ExpandKey извън обработката на данни. При всяка смяна на потребителския ключ се извършва предварително изчисление на всички подключове. Стойностите се записват в оперативната памет и в процеса на криптиране само се прочитат, без да се изчисляват. Много блокове се криптират с един и същ ключ, при което общото време за изпълнение се намалява. Недостатък е заемане на допълнителни 160 байта от оперативната памет за всички подключове.

Декриптирането е в обратен порядък. Първо се извършва ротация на редове надясно (при криптиране е наляво), след което се заменят байтовете според инверсна таблица. Възможно е да се използва една и съща таблица, но това би довело до повече изчисления и

съответно до увеличаване на времето за декриптиране. Търсенето на съвпадение в таблицата е по-бавно и зависимо по време от входните данни. По-високо бързодействие се постига чрез добавяне на нова таблица и увеличаване на програмната памет допълнително с 256 байта. Друга разлика при декриптиране е използването на инверсен полином (3) с коефициенти различни от коефициентите на полинома при криптиране. По-големите коефициенти при декриптиране (E,D,B,9) в сравнение с криптиране (3,2,1,1) увеличават значително времето за изпълнение.

### 3. Резултати

**Таблица 1.** Сравнителни резултати при криптиране и декриптиране с AES-128

	криптиране			декриптиране		
	предварителни изчисления [cycles]	криптиране [cycles]	програма [bytes]	предварителни изчисления [cycles]	декриптиране [cycles]	програма [bytes]
(1) Daeman Rijmen [5]	-	3168	1016	-	-	-
(2) Plos [6]	-	4065	768	-	-	-
(3) Poettering [7]	-	3084	2158	-	4505	-
(3) Poettering [7]	756	2474	1102	1977	3411	2352
	-	4059	-	-	4675	-
(4) Otte [8]	2039	2555	-	-	4764	-
(5) Kim [9]	-	2289	-	-	-	-
(6) получени резултати	782	2328	1165	824	3502	1697
	-	2866	1185	-	-	-

В таблица 1 са приведени получените резултати (6) при софтуерно изпълнение на криптиране и декриптиране с алгоритъм AES-128. Предложените софтуерни решения са сравнени с други известни решения. Най-важният показател за сравнение е необходимият брой цикли за криптиране и декриптиране. На тази основа може да се сравняват различни решения за тяхната ефективност, независимо от честотата на процесора, на който се изпълнява. Друг критерий е необходимият обем памет за запис на програмата. На основата на тези два критерия може да се избере използването на едно, или друго решение. В много случаи намалението на броя цикли е за сметка на увеличението на обема на програмата. Примерно изпълнението на подпрограми намалява обема на използваната памет, но увеличава бързодействието при изпълнение. В първата колонка са разгледани предлагани софтуерни решения и получените резултати. Следващите три колони са за процеса на криптиране и последните три за декриптиране. В първите колонки за двата процеса са брой цикли за предварителни изчисления. При криптиране и декриптиране на блокове от данни се използва един и същ сесиен ключ. Има възможност за предварително изчисление на необходимите подключове в ExpandKey, при което времето за изпълнение в цикъла на Алгоритъм 1 (линии 4 до 7) се съкращава значително. Недостатък е предварителното изчисление и записът в данновата памет, за което се отделят 160 байта при 128 битов ключ. В следващите две колонки са представени необходимият брой цикли за преобразуване съответно за криптиране и декриптиране. Последните две колонки представят необходимия обем памет за запис на програмата. Предложение (1) е на авторите на алгоритъм AES – Joan

Daemen и Vincent Rijmen, разработено на основата на микроконтролер на фирмата INTEL. Авторското предложение (6) е на основата на микроконтролер от фамилия PIC18 на MICROCHIP, докато останалите са на микроконтролери AVR на ATMEL. Предложения (3) и (6) са в два варианта: с предварителни изчисления и без. За предложения (3) и (4) са публикувани и самите програми, което дава възможност за по-детайлни сравнения на показателите и използваните алгоритми. Възможност за намаляване на необходимия брой цикли е при инициализация, да се прехвърли таблицата за замяна S-box от програмната в данновата памет. Характерно за използвания микроконтролер е, че извличането на байт от програмната памет чрез TBLPTR отнема 5 цикъла, докато индиректно адресиране на данновата памет отнема 3 цикъла. Таблицата за замяна S-box се използва за всички 16 байт в SubBytes, което позволява съкращение с  $16 \times 2 \times 10 = 320$  цикъла при преместването на S-box в данновата памет. Предложение (5) е с по-малко на брой цикли от (6), но няма достатъчно данни как е постигнато. Възможно е увеличаване на бързодействието да е постигнато чрез увеличаване на програмната памет или по-продължителни предварителни изчисления.

#### 4. Заключение

В статията се анализира симетричен алгоритъм AES и неговото изпълнение на 8-битов микроконтролер PIC18 на фирмата MICROCHIP. Този алгоритъм намира широко приложение при безжични сензорни мрежи, Smart карти, RFID и др. Детайлно са разгледани четирите обработки на състоянието: SubBytes, ShiftRows, MixColumns и AddRoundKey. За всяка отделна обработка са предложени конкретни решения за подобряване на бързодействието. Тези решения са в резултат на специфичните особености на използвания микроконтролер и на предложения за по-ефективна обработка на данните в процеса по криптиране и декриптиране. В табличен вид е извършен сравнителен анализ на разработеното софтуерно предложение с други публикувани разработки. Основен критерий за сравнение е необходимият брой цикли при криптиране и декриптиране, което показва ефективността на предложените решения. Допълнителни критерии са необходимият обем памет за запис на програмата и времето за извършване на предварителни изчисления. На основата на тези критерии може да се избере за използване едно или друго от представените решения. Предложеното в статията решение е симулирано на програма MPLAB за доказване на правилната обработка на данните и изчисляване на необходимия брой цикли за преобразуване при криптиране и декриптиране.

#### Литература

- [1]. Paar, C., J. Pelzl., T. Guneysu. Understanding Cryptography: From Established Symmetric and Asymmetric Ciphers to Post-Quantum Algorithms, Berlin, Springer-Verlag, 2024.
- [2]. FIPS PUB 197, Advanced Encryption Standard (AES), Federal Information Processing Standards (FIPS), NIST, US Department of Commerce, 2001.
- [3]. Harshali, Z., S. Arbaz. A survey on Various Cryptanalytic Attacks on the AES Algorithm, International journal of Next-Generation Computing, vol. 12, Issue 2, 2021.
- [4]. Biryukov, A., O. Dunkelman, N. Keller, D. Khovratovich, A. Shamir, Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds, EUROCRYPT 2010, LNCS 6110, p.p.299-319.
- [5]. Daeman, J., V. Rijmen, The Block Cipher Rijndael, CARDIS 2000, LNCS 1820, p.p. 277-284, 2000.
- [6]. Plos, T., H. Grob, M. Feldhofer, Implementation of Symmetric Algorithms on a Synthesizable 8-bit Microcontroller Targeting Passive RFID Tags, SAC 2010, LNCS, vol. 6544, p.p. 114-129, Springer Heidelberg (2011).

- [7]. Poettering, B., The AES block cipher on AVR controller, <http://point-at-infinity.org/avraes/>
- [8]. Otte, D., et al., AVR-crypto-lib, [Online], <https://wiki.das-labor.org/w/Crypto-avr-lib/en>
- [9]. Kim, Y., S. Seo, Efficient Implementation of AES and CTR-DRBG on 8-bit AVR-based Sensor Nodes, WISA 2020, Information Security Application, LNCS 12583, p.p. 276-290, 2021.

**За контакти:**

гл. ас. д-р инж. Пламен Й. Стоянов  
катедра „Комуникационна техника и технологии“  
Технически Университет-Варна  
E-mail: pl\_stoianov@tu-varna.bg



# СРАВНЕНИЕ ВРЕМЕТО НА ИЗПЪЛНЕНИЕ НА CRUD ОПЕРАЦИИ ПРИ РЕЛАЦИОННИ И НЕРЕЛАЦИОННИ БАЗИ ОТ ДАННИ

Диян Ж. Динев

**Резюме:** Статията представя резултатите от изследването на времето на изпълнение на CRUD операции при релационни и нерелационни бази от данни. Провежда се теоретичен преглед, който обхваща основните характеристики и различия между двата типа бази от данни в ключови сегменти като структура, мащабируемост, гъвкавост и производителност. Практическата част на анализа се осъществява чрез използване на различни приложения за измерване на ефективността и скоростта на изпълнение на CRUD операциите в разгледаните бази от данни. Въз основа на резултатите от това изследване, статията предоставя препоръки за избора на подходящ тип база от данни според специфичните нужди на различни проекти и приложения.

**Ключови думи:** comparative analysis, MSSQL, MongoDB, non-relational databases, relational databases

## Comparison of execution time of crud operations in relational and non-relational databases

Diyan Zh. Dinev

**Abstract:** The article presents the results of the study of the execution time of CRUD operations in relational and non-relational databases. A theoretical overview is conducted that covers the main characteristics and differences between the two types of databases in key segments such as structure, scalability, flexibility and performance. The practical part of the analysis is carried out by using various applications to measure the efficiency and speed of execution of CRUD operations in the considered databases. Based on the results of this research, the article provides recommendations for choosing an appropriate database type according to the specific needs of different projects and applications.

**Keywords:** comparative analysis, MSSQL, MongoDB, non-relational databases, relational databases

## 1. Въведение

Системите за управление на бази от данни играят ключова роля за ефективното управление на данните, като позволяват на потребителите лесно да извършват различни операции. Подобряването на производителността на тези системи значително повишава ефективността на бизнес процесите и намалява общите разходи. Тези системи предоставят мощен и надежден подход за обработка на множество типове данни. В съвременния свят организациите и компаниите се сблъскват с приложения, които генерират и натрупват огромни количества данни на регулярна основа, водейки до експоненциален ръст на техния обем. Съществуващите релационни бази от данни са широко използвани за тези приложения, но тяхната производителност значително се влошава с увеличаването на обема на данните, като не могат да се справят ефективно с проблема за обработка на големи данни. Освен това, по време на разработването и еволюцията на дадено приложение, релационните бази от данни изпитват затруднения при промяна на схемата във времето, което ограничава тяхната способност да обработват различни типове данни. Тези ограничения водят до въвеждането на NoSQL бази от данни като решение за преодоляване на тези предизвикателства.

NoSQL е въведена не като конкурент на релационните бази от данни, а като алтернативен подход, който адресира проблемите, които релационните бази от данни не могат да решат. NoSQL базите от данни не представляват заместител на релационните бази

от данни, а по-скоро допълнение, което позволява съвместното съществуване на двете технологии. Докато релационните бази от данни се основават на съхраняване на данни в таблици и предварително дефинирани схеми, NoSQL базите от данни не използват или прилагат тези принципи в по-слабо свързан вид. Въпреки че няма универсално или перфектно решение, както релационните, така и нерелационните бази от данни имат своите предимства и недостатъци.

С развитието на ИТ пазара възниква необходимостта от преглед на наличните решения за бази от данни, както и от тяхното адаптиране към различни условия. Настоящата ситуация в ИТ света, заедно с нарастващата популярност на NoSQL базите от данни, насърчава по-задълбочено изследване и анализ на причините, поради които днес се използват все повече нерелационни бази от данни наред със стандартните релационни бази от данни. В тази статия са представени релационните и нерелационните модели на бази от данни, както и анализ времето на изпълнение на CRUD операции при двата вида бази от данни – релационни и нерелационни.

## 2. Релационни бази от данни

Релационните бази от данни (RDBMS) са основни компоненти в управлението на данни от десетилетия насам. Те се основават на релационния модел, предложен от Едгар Код през 1970 г., който структурира данните във форми на таблици (релации), състоящи се от редове и колони. Тази структура позволява ясна организация на данните и улеснява достъпа до тях чрез езика за заявки SQL (Structured Query Language). Всяка таблица съдържа уникален ключ, който идентифицира всеки ред, а връзките между таблиците се осъществяват чрез използването на първични и външни ключове [1][2].

Основни характеристики на релационните бази от данни [2][3]:

1. Строга схема на данните: Релационните бази от данни разчитат на предварително дефинирани схеми, което осигурява последователност и валидност на въведените данни. Всяка таблица има ясно дефинирана структура и типове данни за всяка колона.

2. Свойства: Те поддържат транзакции, които гарантират Atomicity (атомарност), Consistency (последователност), Isolation (изолация) и Durability (устойчивост). Това е особено важно за критични бизнес приложения, където загубата на данни или непоследователностите не са допустими.

3. Надеждност и консистентност: RDBMS системите осигуряват висока степен на надеждност чрез интеграцията на системи за възстановяване на данни и резервиране. Строгото прилагане на ограниченията за целостта на данните (например уникални и външни ключове) гарантира, че данните остават точни и консистентни.

4. Сложни заявки: SQL предоставя мощни възможности за изпълнение на сложни заявки с помощта на JOIN операции, подзаявки и агрегации. Това прави релационните бази от данни подходящи за приложения, изискващи подробни анализи и репорти.

5. Подходящи за OLTP: Релационните бази от данни се използват широко в транзакционни системи (OLTP - Online Transaction Processing), където бързото и надеждно управление на данни е критично за бизнес процесите.

Предимства:

- Структурирани данни: Идеални за системи, които боравят с високо структурирани данни, където формата на данните рядко се променя.

- SQL като стандартен език: SQL е стандартизиран и широко разпространен, което улеснява разработчиците и администраторите при работа с различни RDBMS системи.

- Целостта на данните: Стриктните схеми и ограниченията на базите от данни гарантират, че данните остават консистентни и точни.

Недостатъци:

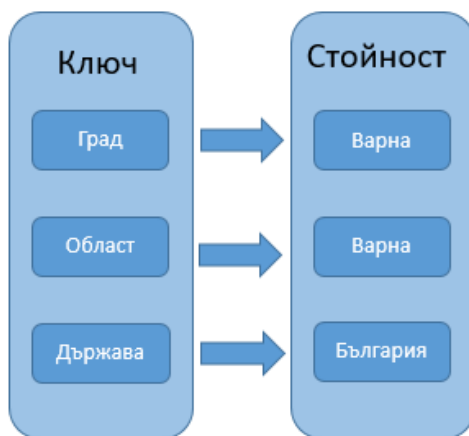
- Лоша мащабируемост: При мащабиране по хоризонтала (разширяване на системата чрез добавяне на повече сървъри) релационните бази от данни срещат сериозни затруднения, особено при обработка на големи обеми данни.
- Не гъвкави спрямо неструктурирани данни: Релационните бази от данни не са подходящи за обработка на неструктурирани или полуструктурирани данни, като документи, JSON или графови структури.
- Сложни и скъпи актуализации на схемата: Промяната на структурата на базата от данни (например добавяне на нови колони или промяна на типа данни) може да изисква значителни усилия и ресурси.

### 3. Видове нерелационни бази от данни

В тази част са разгледани основни типове NoSQL хранилища на данни и популярен пример за всеки от тях.

#### А. Хранилища с ключ-стойност

Използването на хранилища с ключ-стойност означава, че съхранените стойности са свързани с конкретен ключ и единственият подходящ начин за извличане на данни е чрез този ключ. Тези хранилища използват структура на данните, подобна на тази в карти (maps) и речници (dictionaries), където данните могат да бъдат манипулирани и обработвани чрез уникален ключ [4]. Гъвкавостта на тези хранилища ги прави удобни за съхраняване на данни в неструктуриран формат. Освен това те позволяват бързи и мащабируеми произволни заявки за четене/запис, както и ефективно извличане на поисканите данни [5]. Тези хранилища се използват от Facebook за съхранение на публикации с уникални идентификатори. Стойността на даден уникален идентификатор съдържа съобщението, идентичността на потребителя и времето на публикуване на съответната публикация [6]. Хранилищата с ключ-стойност са подходящи в случаите, когато трябва да се съхрани сесията на потребителя, количката за пазаруване или да се получи информация за любимите продукти на потребителя. Фигура 1 илюстрира прост пример за структура на данни в хранилище с ключ-стойност.



Фиг. 1. Проста структура на данните на хранилище за данни ключ-стойност

Едно от най-популярните хранилища с ключ-стойност е Redis, разработено от Салваторе Санфилипо [7]. Това хранилище с отворен код има способността да осигурява бързи и мащабируеми произволни заявки за четене/запис. То може да обработва повече от 100 000 операции за четене или запис в секунда. Redis също така поддържа различни типове

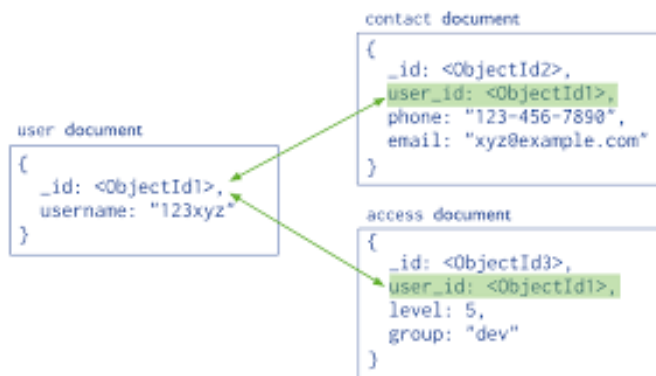


структури от данни, като низове, хешове, списъци, множества, подредени множества, битмапи и геопространствени индекси. Освен това разполага с вградени реплики, които могат да се репликират чрез модел master-slave, при който един master може да има множество slaves [8].

## Б. Document Datastores

Хранилищата за документи се използват за съхранение и организиране на данни под формата на документи. Тези документи позволяват съхранение и извличане на данни в различни формати като XML (Extensible Markup Language), PDF и JSON (JavaScript Object Notation). Хранилищата за документи са много гъвкави по своята същност, тъй като не изискват предварително дефинирани схеми. Те също така се характеризират с възможността да добавят голям брой различни полета към един или повече документи, без да се губи място за добавяне на същите празни полета към други документи [9][10]. Документите се групират заедно в колекции. Въпреки че една колекция е съставена от много документи, всеки документ може да има различна схема и различни типове съхранени данни. Всеки документ притежава уникален идентификатор в рамките на своята колекция. Хранилищата за документи са подходящи за уеб приложения, които включват съхранение на полуструктурирани данни и изпълнение на динамични заявки. Фигура 2 изобразява прост пример за структура на данни в хранилище за документи.

MongoDB е едно от най-популярните хранилища за документи с отворен код, написано на езика за програмиране C++ и разработено от софтуерната компания 10gen [11]. Това е високопроизводително и ефективно хранилище за данни. То е също така гъвкаво и без схеми, като може да включва една или повече колекции от документи. MongoDB може да се използва за съхранение и персонализиране на големи файлове като изображения и видеоклипове. Също така разполага със сложен език за заявки и поддържа MapReduce за обработка на разпределени данни [12].



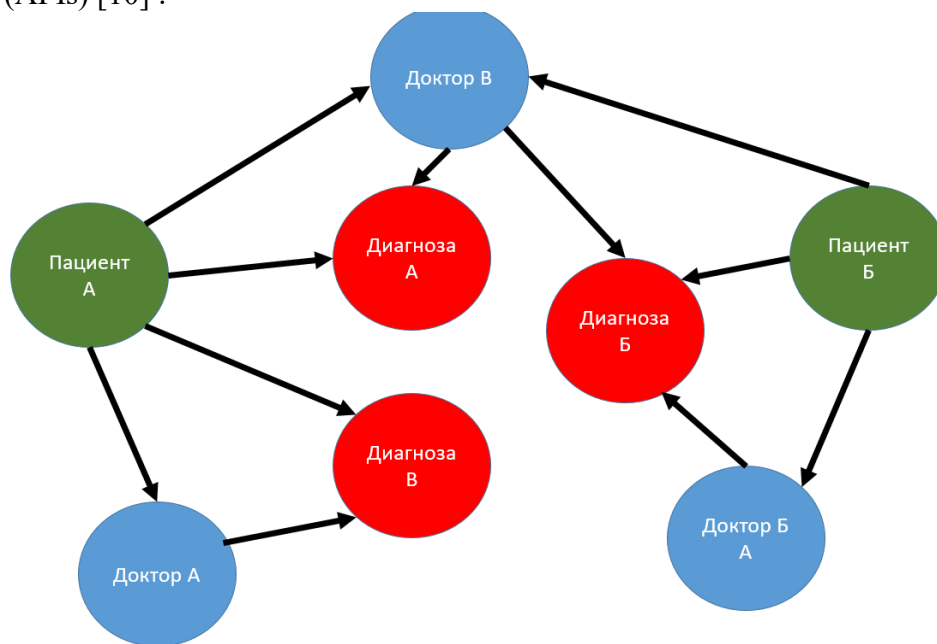
Фиг. 2. Проста структура на данните на хранилище за данни от тип документи

## В. Граф базирани хранилища на данни

Графовите хранилища на данни са проектирани около идеята за графова структура, която съдържа възли, свойства и свързващи ръбове. Възлите представляват обекти, свойствата описват реална информация за обектите, а ръбовете представляват връзките между възлите. Графовите хранилища използват усъвършенствани алгоритми за най-кратък път, за да направят процеса на заявяване на данни по-ефективен. Повечето от тези хранилища са без схема и малко от тях поддържат хоризонтално мащабиране, тъй като е трудно да се преминава и манипулира граф, когато свързаните възли са разпределени по клъстери. Графовите бази данни са специализирани в решаването на проблеми с намирането

на пътища в навигационни системи [13]. Те също така са проектирани да бъдат подходящи за представяне на силно свързани данни като социални връзки, географски данни, социални мрежи, биоинформатика и управление на облаци [14].

Фигура 3 показва прост пример за структура на данни на графово хранилище с неговите възли и насочени ръбове. Тя показва няколко потребители, какво харесват и кого следват. Neo4j е една от най-популярните и мощни графови бази данни, написана на Java. Това е високопроизводителна графова база данни, която може да осигури гъвкава мрежова структура. Тя е високо достъпна и мащабируема, тъй като има способността да съхранява и организира огромен брой възли и връзки между тях ефективно. Тя разполага с език за заявки Cypher, който се използва за бързо заявяване и ефективно преминаване. Тя също така предлага интерфейс за представителен държавен трансфер (REST) и Java програмни интерфейси (APIs) [10].



Фиг. 3. Структура на данните от тип Graph

#### 4. Сравнение на времето за изпълнение на CRUD операции, използвайки MSSQL и MongoDB бази от данни

Възможността за ефективно управление на данни е от съществено значение за съвременните приложения, което налага необходимостта от избор на подходяща база от данни.

CRUD операцията (Create, Read, Update, Delete) представляват основните действия, извършвани при работа с данни, и тяхното времево изпълнение е ключов фактор за производителността на приложението.

MSSQL и MongoDB са две от най-популярните системи за управление на бази от данни, всяка от които предлага уникални характеристики и предимства за различни сценарии на употреба. Сравнението на времето за изпълнение на CRUD операции между тези две бази данни ще предостави важна информация относно тяхната производителност и ефективност при обработка на данни.

Целта на настоящото сравнение е да анализира и сравни времето за изпълнение на CRUD операции, използвайки MSSQL и MongoDB, с акцент върху различията в тяхната производителност и приложимост за различни типове проекти. Всички заявки и тестове за

производителност са изпълнени на една и съща компютърна конфигурация. Таблица 1 показва примерни заявки за изпълнение при MSSQL и MongoDB.

**Таблица 1.** Примерни заявки за изпълнение

MSSQL	MongoDB
SELECT * FROM Film	db.Book.find();
SELECT B.Book_Name, B.Release_year, G.Genre_name FROM Books as B JOIN Book_Genre as BG ON BG.BookId = B.BookId JOIN Genre AS G ON BG.GenreId = G.GenreId	db.Book.find();
SELECT B.Book_Name, B.Release_year, G.Genre_name FROM Books as B JOIN Book_Genre as BG ON BG.BookId = B.BookId JOIN Genre AS G ON BG.GenreId = G.GenreId WHERE B.Book_name = 'Под Игото'	db.Book.find({'Book_name':'Под Игото'});
SELECT COUNT(*) as BookCount FROM Books	db.Book.count()

Таблица 2 показва резултатите от времето за изпълнение на заявки с различна сложност в MSSQL базата данни и MongoDB базата от данни. Данните в двете бази данни са идентични. Броят на записите, създадени в таблицата с „Книги“, е 100 000. Следва да се подчертае, че много често резултатите от изпълнението на заявка и в двете бази от данни са значително по-бавни при първото изпълнение; когато заявката се изпълни отново, изпълнението е значително по-бързо. Резултатите, показани в таблицата, са взети като средна стойност от пет последователни измервания върху същия брой записи. Фигура 4 показва сравнение на времето за изпълнение на заявките при MSSQL и MONGODB.

**Таблица 2.** Време за изпълнение на заявките

Време за изпълнение на MSSQL заявка	Време за изпълнение на MongoDB заявка	Успешно върнати записи
389 ms	50ms	100 000
468 ms	49ms	100 000
12ms	19ms	1
10ms	16ms	1



**Фиг. 4** Сравнение при времето за изпълнение на заявките при MSSQL и MongoDB.

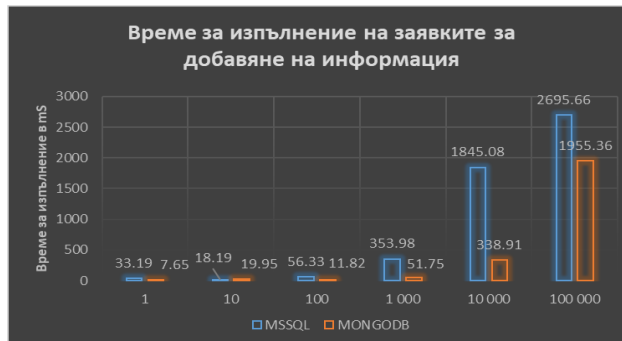
За да се измери и сравни времето за изпълнение на основни CRUD операции върху MS SQL бази данни и MongoDB бази от данни, е създадено сравнително опростено приложение, в което са направени връзки към двете бази от данни. Приложението е разработено на програмния език C#. Достъпът до базите от данни е осигурен чрез NuGet пакетите MongoDB.Driver и MongoDB.BSON. Като се има предвид, че са използвани две бази от данни, за да се осигури бърза и лесна смяна на базите от данни, низовете за връзка, съдържащи пътищата към базите от данни, са съхранени в конфигурационен файл. В самия код няма значителни разлики между начините на използване на двете бази от данни. Стъпките при създаването на проекта включват създаване на връзка, създаване на методи за добавяне на данни в базите от данни, четене, модифициране и изтриване (т.нар. CRUD операции). Създадени са методи, които генерират набори от данни в зависимост от предадения параметър за брой записи. За целите на тестването е създаден тестов набор от

данни. CRUD операциите върху данните са изпълнени върху проби от 1, 10, 100, 1000, 10,000, 100,000 записа, за да се наблюдава зависимостта на времето с увеличаването на броя на записите.

Таблица 3 показва времето за изпълнение на операция добавяне при MSSQL и MongoDB, а на фигура 5 е показано тяхното графично представяне.

**Таблица 3.** Време за изпълнение на заявките за добавяне на информация

Време за изпълнение на MSSQL заявка	Време за изпълнение на MongoDB заявка	Успешно добавени записи
33.19 ms	7.65 ms	1
18.65 ms	19.95 ms	10
56.33 ms	11.82 ms	100
353.98 ms	51.75 ms	1 000
1845.08 ms	338.91 ms	10 000
2695.66 ms	1955.36 ms	100 000

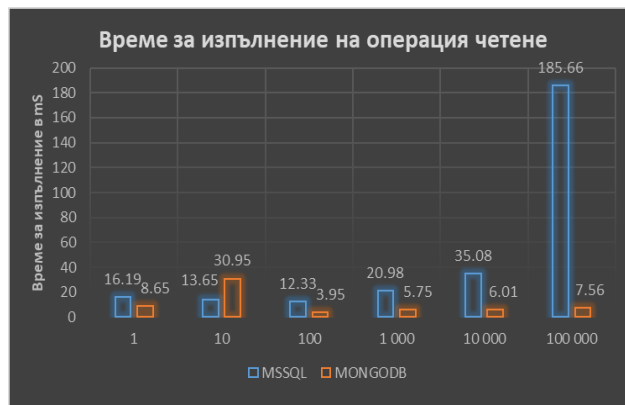


**Фиг. 5.** Сравнение при времето за изпълнение на операция добавяне при MSSQL и MongoDB.

Операцията по четене от база от данни се влияе от множество фактори, като съществено значение имат структурата на данните и начинът на тяхното съхранение. Поради неструктурираната си природа, методът на извличане на данни в MongoDB може да бъде сложен, особено при търсене и намиране на информация. Времето за изпълнение на операция по четене зависи главно от сложността на структурата на данните и начина, по който те са записани. В конкретното изследване използваните данни не са със сложна структура, което води до по-добра производителност на MongoDB при извличане на прости записи в сравнение с MSSQL. Резултатите от експериментите показват, че времето за изпълнение на заявки в MongoDB е по-кратко, особено при работа с голям брой прости записи. Измерванията се базират на средното време за извличане на данни при различни обеми от записи. Резултатите от измерванията са представени в Таблица 4, и визуализирани на фигура 6.

**Таблица 4.** Време, необходимо за изпълнение на операция четене.

Време за изпълнение на MSSQL заявка	Време за изпълнение на MongoDB заявка	Успешно прочетени записи
16.19 ms	8.65 ms	1
13.65 ms	3.95 ms	10
12.33 ms	3.82 ms	100
20.98 ms	5.75 ms	1 000
35.08 ms	6,01 ms	10 000
185.66 ms	7.56 ms	100 000



**Фиг. 6.** Време за изпълнение на операция четене при MSSQL и MongoDB.

**Таблица 5.** Време, необходимо за изпълнение на операция четене

Време за изпълнение на MSSQL заявка	Време за изпълнение на MongoDB заявка	Успешно редактирани записи
17.12 ms	25.05 ms	1
15.66 ms	10.05 ms	10
15.68 ms	11.82 ms	100
124.98 ms	64.95 ms	1 000
844.08 ms	618,06 ms	10 000
7195.66 ms	5411.36 ms	100 000



**Фиг. 7.** Време за изпълнение на операция промяна при MSSQL и MongoDB

Промените в данните на базата от данни могат да бъдат извършени по различни критерии. При изпълнението на заявката в приложението е използвано актуализиране на всички записи, като са модифицирани два атрибута в рамките на всеки запис. Производителността на изпълнението зависи от критериите, използвани за намиране на конкретен запис. По-добри резултати се наблюдават при MongoDB, докато при сложни критерии MSSQL демонстрира по-добра ефективност. Например, при промяна на всички записи, които съдържат стрингови данни „Варна“, в полето за населено място. Операциите по промяна на записите и сравнението на времето за тяхното изпълнение са представени в таблица 5 и фигура 7.

Производителността на операциите по изтриване в двете бази от данни са показани в таблица 6 и показват сходни времена за изпълнение. При изтриване на 10 000 записа се наблюдава по-добро средно време за изпълнение при MSSQL сървъра, докато в други случаи времето за изпълнение е приблизително еднакво. Това е представено визуално на фигура 8.

**Таблица 6.** Време, необходимо за изпълнение на операция изтриване

Време за изпълнение на MSSQL заявка	Време за изпълнение на MongoDB заявка	Успешно изтрити записи
18.95 ms	9.55 ms	1
16.97 ms	8.15 ms	10
12.98 ms	5.52 ms	100
22.08 ms	46.38 ms	1 000
157.68 ms	429,06 ms	10 000
1505.63 ms	1495.36 ms	100 000



**Фиг. 8.** Време за изпълнение на операция изтриване при MSSQL и MongoDB7.

## Заклучение

Проведените изследвания в областта на базите от данни по време на подготовката на документа и анализът на получените резултати показват, че MongoDB демонстрира по-добра производителност при по-голям обем записи и данни. При до 1000 записа времената за изпълнение са сходни, но при повече от 1000 записа MongoDB значително превъзхожда. В случаите, когато MSSQL показва по-добри резултати, се установява, че използването на MSSQL Management Studio директно върху базата от данни води до по-кратки времена за

изпълнение. Следователно MSSQL базите от данни са по-подходящи за малки и средни приложения, особено когато производителността не е основен приоритет. Релационните бази от данни се използват масово и предоставят добри резултати при работа с ограничен обем данни. Изборът на база от данни трябва да бъде обмислен, като се вземат предвид ключови фактори като обем на данните, гъвкавост, схема, бюджет, тип на сървъра, количество и честота на транзакциите. Тези фактори не са единствените, тъй като изборът зависи също и от спецификата на компанията и целта на разработваното приложение.

### Литература

- [1]. Silberschatz A., Korth H. F., and Sudarshan S.. 2019. "Database System Concepts" (7th ed.). McGraw-Hill Education.
- [2]. Odhiambo S., De Raeve A. , Copot C., Radulescu I. R., Rudolf A., Penko T., and Cools J.. 2024. "Creation of Databases for a Virtual Training Library in Fashion Design." Communications in Development and Assembling of Textile Products, 5(2), 140–150. <https://doi.org/10.25367/cdatp.2024.5.p140-150>.
- [3]. Prakash C., Dhyani B., and Lalit N.. 2024. "A Comprehensive Study on Structural and Non-Structural Databases and Its Impact on Hybrid Databases." In G. Lakhera, S. Taneja, E. Ozen, M. Kukreti, and P. Kumar (Eds.), Navigating the World of Deepfake Technology (pp. 351-377). IGI Global. <https://doi.org/10.4018/979-8-3693-5298-4.ch018>.
- [4]. Makris A., Tserpes K., Anagnostopoulos D, and Andronikou V., "A classification of NoSQL data stores based on key design characteristics," in Proc. 2nd International Conference on Cloud Forward: From Distributed to Complete Computing, Madrid, Spain, 2016, pp. 94-103.
- [5]. Bhogal J., and Choksi I, "Handling big data using NoSQL," in Proc. IEEE Conference on Advanced Information Networking and Applications Workshop, Gwangju, Korea, 2015, pp. 393-398.
- [6]. Srivastava P., Goyal S., and Kumar A., "Analysis of various nosql database," in Proc. 2015 IEEE International Conference on Green Computing and Internet of Things (ICGCIoT), Nodia, India, 2015, pp. 539-544.
- [7]. Redis Labs, "Redis FAQ" Internet: <http://redis.io/topics/faq>, Last visit 10.10.2024.
- [8]. Redis, <http://redis.io/http://www.neo4j.org/>, Last visit 10.10.2024
- [9]. Deka Г., "Fine A Survey of Cloud Database Systems," in IT Professional, vol. 16, no.2, pp. 50-57, 03 January 2013.
- [10]. Kaur K., and R. Rani R., "Modeling and querying data in nosql databases," in Proc. 2013 International Conference on IEEE on Big Data, Silicon Valley, CA, USA , Oct. 2013, pp. 1-7.
- [11]. Padhy R., Patra M., and Satapathy S., "RDBMS to NoSQL: Reviewing some next-generation nonrelational database's," International Journal of Advanced Engineering Sciences and Technologies, Vol. 11, PP.15-30, 2011.
- [12]. Miller J., "Editor Graph database applications and concepts with Neo4j," in Proc. 23rd-24th Southern Association for Information Systems Conference, Atlanta, GA, USA, 2013, pp.141-147.
- [13]. Angadi A., and Gull K., "Growth of new databases & analysis of NoSQL datastores," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, pp. 1307-1319, 2013.
- [14]. Angles R., and Gutierrez C., "Survey of graph database models" ACM Computing Survy, no.1, pp.1-39, Feb. 2008.

#### За контакти:

гл. ас. д-р инж. Диян Желев Динев  
катедра „Софтуерни и Интернет Технологии“  
Технически университет – Варна  
E-mail: [diyandinev@tu-varna.bg](mailto:diyandinev@tu-varna.bg)

# INTEGRATING ADAPTIVE AUTOSAR WITH SERVICE-ORIENTED ARCHITECTURES IN AUTOMOTIVE SYSTEMS

Ivan Orfey Ivanov

**Abstract:** The automotive industry is rapidly evolving with the integration of advanced software architectures. Adaptive AUTOSAR and Service-Oriented Architectures (SOA) are at the forefront of this transformation, offering enhanced flexibility, scalability, and interoperability for modern vehicle systems. This paper explores the integration of Adaptive AUTOSAR with SOA, discussing the benefits, challenges, and potential solutions. Key technologies, including SomeIP and middleware solutions, are examined to provide a comprehensive understanding of the current state and future directions in automotive software development.

**Keywords:** Adaptive AUTOSAR, Service-Oriented Architecture, SOA, SomeIP, automotive systems, middleware, vehicle communication

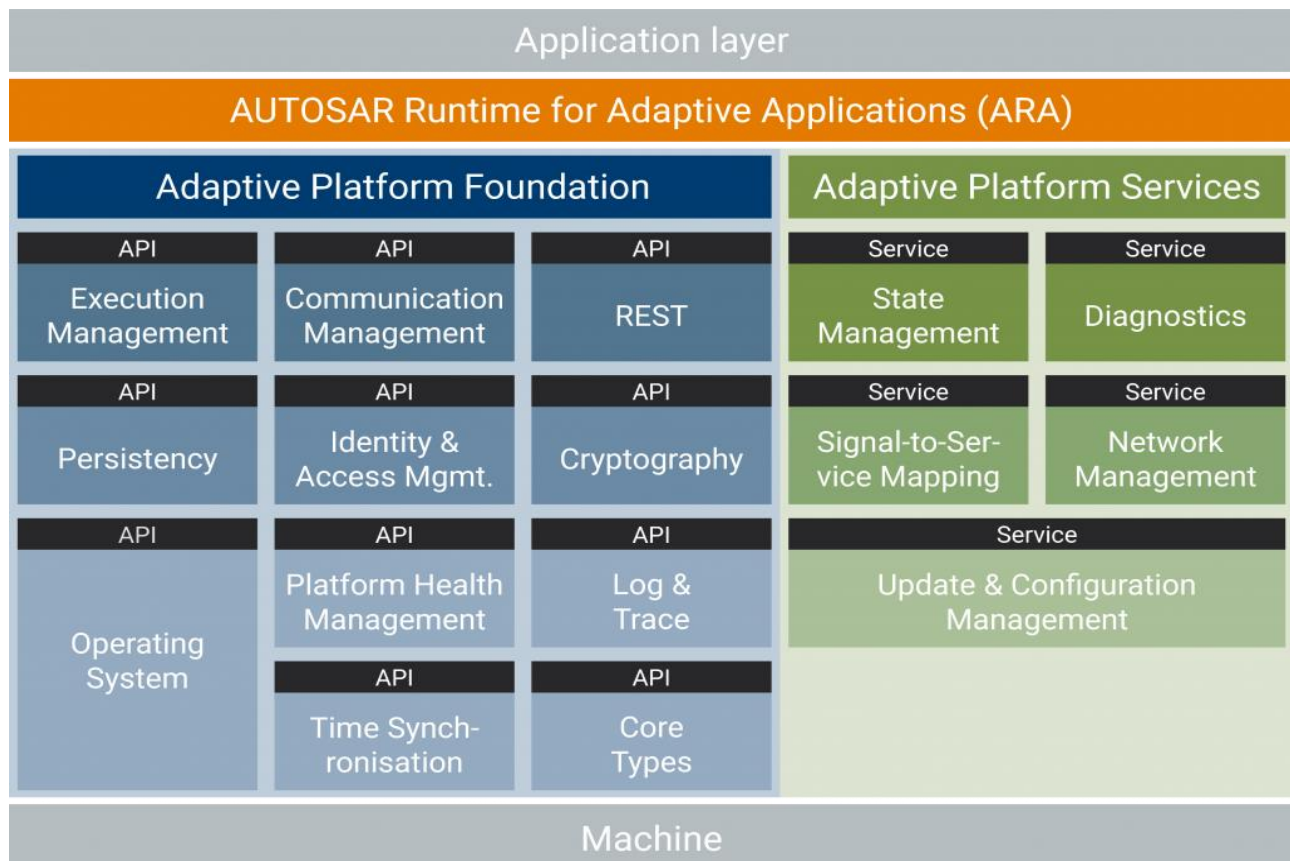
## 1. Introduction

Historically, one of the main factors driving the automotive industry has been customer expectations and demands, which subsequently become trends that all OEMs must comply with in order to be competitive in the market. The extremely intensive development of information technology in recent decades has confirmed some indisputable trends that lead car manufacturers to provide technological innovations that satisfy the customer's requirements. Easily noticeable examples of such trends are autonomous driving, electrification, connectivity, which has made the topic of receiving functional improvements through over-the-air (OTA) updates in modern vehicles increasingly popular. These aspects are changing the way we perceive modern cars as software-defined vehicles (SDVs), where the separation of hardware and software is not only possible, but highly desirable, which can be achieved thanks to Service Oriented Architecture (SOA). This allows software components to be developed, updated and deployed independently, which is critical for SDVs where functionalities are highly dependent on software services. All those actual directions in the automotive area require much more software which reveals some big challenges to face on. To address these evolving demands, Adaptive AUTOSAR has emerged as a crucial standard that is designed to complement the principles of SOA, provides a flexible and dynamic framework that supports the integration of complex software systems required for modern automotive applications. By leveraging Adaptive AUTOSAR, manufacturers can ensure that their vehicles can handle the sophisticated computing tasks associated with autonomous driving, electrification, and V2X (Vehicle-to-Everything) communication. Furthermore, the modularity provided by the standard allows for seamless integration of new features and services, enhancing the vehicle's functionality over its lifespan. The convergence of Adaptive AUTOSAR and Service-oriented Architecture can be considered as a significant advancement in the automotive industry which gives the opportunity for comprehensive infrastructure that supports the rapid development, deployment, and updating of software, ultimately leading to smarter, more adaptable, and more reliable vehicles.

## 2. Examination of the AUTOSAR Adaptive Platform

What can be noticed in the depicted Figure 1 below is an architecture logical view of the Adaptive Platform (AP), which gives good understanding of what layers it is divided. Starting from

the top, the Application layer takes its place, constructed by all Adaptive Applications developed by the automotive engineers. Those applications are running on top of ARA which stands for AUTOSAR Runtime for Adaptive Applications. This layer of the AP contains application interfaces provided by Functional Clusters, which belong to either Adaptive Platform Foundation shown on the left side of the figure or Adaptive Platform Services on its right side. Functional clusters play a crucial role in the AUTOSAR Adaptive Platform by organizing and managing various functionalities, some of which are: modular organization, service provisioning and resource management. Finally, POSIX PSE51 can be mentioned that provides a standardized, efficient, and real-time capable environment that aligns with the objectives of the AUTOSAR Adaptive Platform.



**Fig. 1.** Architecture logical view of Adaptive Platform

Adaptive AUTOSAR applications are organized in software components (SWCs) that communicate via services that they may provide or request. We call a SWC that provides a service a server and an SWC that requests a service a client. Client and server roles may be fulfilled by the same SWC. SWCs provide or request services as needed; the binding between clients and servers is determined at runtime by the middleware through service discovery. The dynamic binding of services is the core mechanism for providing adaptivity in AP [1].

The typical development process for the Adaptive AUTOSAR Platform is to create an ARXML Manifest file and design an AUTOSAR modeling and E/E architecture based on the Manifest file. Typically, this is completed with a toolchain from companies that offer different AUTOSAR tools. The specification for the application to be executed is written and passed as an argument through the ARXML Manifest file, and the argument is then used to execute the Adaptive AUTOSAR Platform. First, Execution Management via `ara::exec` is executed, and then the AA, which is responsible for interfacing with Execution Management, is executed [2].



### 3. Methodology and Technical Scope

There are two major groups of technology drivers behind the Adaptive Platform. One is Ethernet, and the other is processors. The ever-increasing bandwidth requirement of the on-vehicle network has led to the introduction of Ethernet, that offers higher bandwidth and with switched networks, enabling the more efficient transfer of long messages, point-to-point communications, among others, compared to the legacy in-vehicle communication technologies such as CAN. The Classic Platform (CP), although it supports Ethernet, is primarily designed for the legacy communication technologies, and it has been optimized for such, and it is difficult to fully utilize and benefit from the capability of Ethernet-based communications. On the other hand, the AUTOSAR Adaptive Platform is specifically designed to take full advantage of Ethernet's high bandwidth, switched network capabilities, offering a more suitable architecture for modern vehicles compared to the CP. Similarly, performance requirements for processors have grown tremendously in recent years as vehicles are becoming even more intelligent.

It is also worthwhile to mention that there is a combined effect of both processors and faster communications. As more processing elements are being combined in a single chip like manycore processors, the communication between these processing elements is becoming orders of magnitude faster and efficient than legacy inter-ECU communications. This has been made possible by the new type of processor interconnect technologies such as Network-on-Chip (NoC). Such combined effects of more processing power and faster communication within a chip also prompts the need for a new platform that can scale over ever-increasing system requirements [3].

Some key features of Ethernet in the scope of Automotive Systems are mentioned and explained in Table 1.

**Table 1.** Summary of Ethernet in Automotive Industry

Feature	Description
Technology Type	High-speed, packet-based communication protocol
Bandwidth	Typically ranges from 100 Mbps to 1 Gbps or higher
Network Topology	Commonly used in star or switched network topologies for efficient data handling
Use Cases	Infotainment systems, ADAS (Advanced Driver Assistance Systems), V2X (Vehicle-to-Everything) communication, over-the-air (OTA) updates
Real-Time Capabilities	Supports real-time communication with protocols like TSN (Time-Sensitive Networking) for deterministic data transfer
Security	Includes various security measures such as encryption and secure protocols to protect data integrity and confidentiality

The modeling of service-oriented E/E architectures and the lower-level Ethernet networks is already relevant for many ECUs of different performance classes. The spectrum extends from intelligent sensors such as camera, radar and LIDAR systems through to central high-performance computers that implement sensor fusion and the higher-level driver assistance applications. ECUs in the lower and medium performance classes are often based on a conventional AUTOSAR software architecture. However, other operating systems are necessary for high-performance ECUs. On the one hand, support is needed for the corresponding hardware accelerators, for example for graphical units. On the other hand, there is a need for greater flexibility in application handling and the updating of data. This is why POSIX-based operating systems are mostly used in this field. The best-

known software solutions in the automotive environment include Linux – for safety-relevant applications under Linux [4].

#### 4. Overview of SOME/IP Protocol Specification

In the context of integrating service-oriented architectures within automotive systems, the Service-Oriented Middleware over IP (SOME/IP) plays major role. SOME/IP is a communication protocol that facilitates efficient and fluid service-oriented communication between software components in modern vehicles. It is designed to support the complex requirements of in vehicle networks, enabling seamless data exchange over Ethernet networks. This protocol aligns with the adaptive platform's goals by supporting dynamic discovery and binding of services, allowing for more modular and expandable system designs. The integration of SOME/IP into the AUTOSAR Adaptive Platform helps in achieving interoperability and coherent communication across various vehicle domains, leveraging the high bandwidth and low latency of Ethernet networks. SOME/IP is flexible and scalable, supporting various communication methods and data types. In addition, it has many other advantages, like not depending on the platform and being economical because it transmits data only when the receiver needs the data. Since the data exchanged through SOME/IP in the vehicle are mainly vehicle control messages, it cannot be emphasized enough how vital SOME/IP communication security is. A vehicle control message that has been maliciously manipulated by an attacker causes the vehicle's control system to operate in a way the driver does not want, which can lead to life-threatening consequences for the driver and passengers [5].

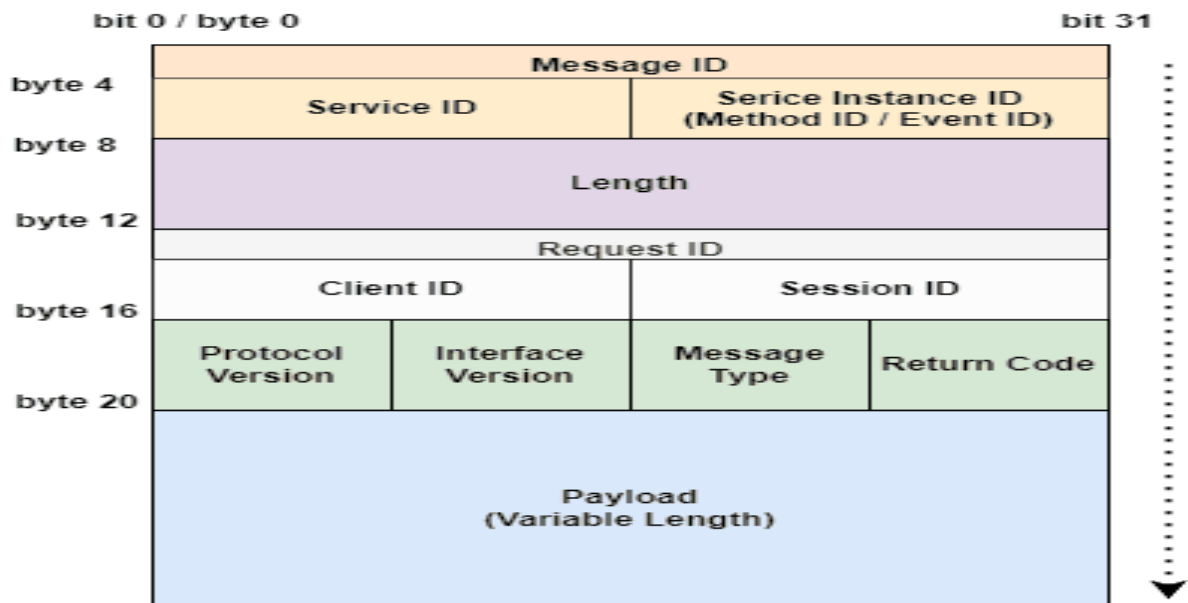
SOME/IP is based on service definitions that list the functionality that the service provides. A service can consist of combinations of zero or multiple events, methods and fields. Events provide data that are sent cyclically or on change from the provider to the subscriber. Methods allow the subscriber to issue remote procedure calls (RPCs) executed on provider side. Fields are combinations of one or more of the following three:

- a notifier which sends data on change from the provider to the subscribers
- a getter which can be called by the subscriber to explicitly query the provider for the value
- a setter which can be called by the subscriber when it wants to change the value on provider side

The major difference between the notifier of a field and an event is that events are only sent on change, the notifier of a field additionally sends the data directly after subscription [6].

On Figure 2 below, the SOME/IP protocol data frame structure is seen. It is mainly composed of the message header (Header) and message body (Payload). The Header part contains:

- Message ID which shall be a 32-Bit identifier that is used to identify the RPC call to a method of an application or to identify an event.
- Length field shall contain the length in Byte starting from Request ID/Client ID until the end of the SOME/IP message.
- Request ID - The Request ID allows a server and client to differentiate multiple parallel uses of the same method, getter or setter.
- Protocol Version identifying the used SOME/IP Header format.
- Interface Version shall be an 8-Bit field that contains the Major Version of the Service Interface.
- Message Type is used to differentiate different types of messages.
- Return Code shall be used to signal whether a request was successfully processed.



**Fig. 2.** Structure of SOME/IP data frame

By following the explanations regarding the different fields mentioned above and as well as using the formula below, the total size of a data frame in the SOME/IP protocol can be calculated by summing the sizes of the header and the payload.

$$\text{Frame size} = \text{Header size} + \text{Payload size} \quad (1)$$

In order to ensure the interoperability of the SOME/IP protocol, the format of all protocol headers is consistent. The fields in the header are displayed according to the transmission order. The fields in the upper left corner are first sent, and the order to be processed is also determined by the position of the fields in the message. Therefore, unicast, multicast, and broadcast are supported, and different communication types can be applied to different communication scenarios [7].

A crucial aspect of the SOME/IP protocol is its robust Service Discovery mechanism, which plays a pivotal role in enabling dynamic and efficient communication within service-oriented automotive architectures. Service Discovery in SOME/IP allows for the automatic detection, announcement, and connection of services between software components across the vehicle's network. This functionality is essential in modern automotive systems, where software components must dynamically interact with one another to provide advanced features and adapt to varying operational conditions. By leveraging Service Discovery, the AUTOSAR Adaptive Platform can seamlessly manage the lifecycle of services, ensuring that communication endpoints are consistently and reliably established across the vehicle's Ethernet network.

SOME/IP-SD (Scalable Service-Oriented MiddlewarE over IP - Service Discovery) is used to:

- Locate service instances
- Detect if service instances are running
- Implement the Publish/Subscribe handling

Inside the vehicular network service instance locations are commonly known; therefore, the state of the service instance is of primary concern. The location of the service (i.e. IP-Address, transport protocol, and port number) are of secondary concern [8].

The service discovery process (Figure 3) is performed when the system starts, including three phases: initial wait, repetition, and main. Servers and clients notify each other of service

information through SOME/IP-SD messages, consisting of the entries array and options array [9]. The find service entry shall be used for finding service instances from the client side and shall only be sent if the current state of a service is unknown, while the offer service entry is used by the server side to explicitly offer access to particular service to the given client. Next step performed by the client is to subscribe for receiving events part of the provided service or event group containing specific events and on the other hand the server side has to acknowledge the subscription. After that, the communication for sending and receiving data between the client and server starts.



**Fig. 3.** Sequence of service discovery process

## 5. Conclusion

This study mentions some popular trends in the automotive industry and address the accompanying challenges with them. The research goes through one major problem that car manufactures are forced to deal with which is mainly the growing amount and complexity of software in modern vehicles. It suggests integrating Adaptive AUTOSAR with service-oriented architectures as an effective engineering approach to face this issue, highlighting the benefits of Adaptive Platform and its compatibility with Ethernet-based communications in vehicle networks, as well as the usage and specifics of the SOME/IP communication protocol.

## References

- [1]. Menard, Christian, Goens, Andrés, Lohstroh, Marten, Castrillón, Jerónimo. Achieving Determinism in Adaptive AUTOSAR. Grenoble: The Design, Automation, and Test in Europe (DATE) Conference and Exhibition, March 9-13 2020, pp. 3-5
- [2]. Hong, Dongwon, Moon, Changjoo. Autonomous Driving System Architecture with Integrated ROS2 and Adaptive AUTOSAR. Electronics: March 2024, pp. 7-9
- [3]. AUTOSAR. Explanation of Adaptive Platform Design. AUTOSAR AP R22-11 REL: 29 March 2019, pp. 12-18
- [4]. Jesse, Bernd, Weber, Marc, Helmling, Markus. The Future with SOA, POSIX, TSN. Automotive Ethernet: Trends and Challenge. Automobil-Elektronik: 12 December 2017, pp. 2-4
- [5]. Lee, Seulhui, Choi, Wonsuk, Lee, Dong Hoon. Protecting SOME/IP Communication via Authentication Ticket. Sensors: 11 July 2023, pp. 1-3

- [6]. AUTOSAR. SOME/IP Protocol Specification. AUTOSAR FO R22-11 REL: 29 March 2019, pp. 18-25
- [7]. Du, Jinze, Tang, Rui, Feng, Tao. Security Analysis and Improvement of Vehicle Ethernet SOME/IP Protocol. Sensors: 8 September 2022, pp. 4-6
- [8]. AUTSOAR. SOME/IP Service Discovery Protocol Specification. AUTOSAR FO R20-11 REL: 30 November 2020, pp. 18-23
- [9]. Luo, Feng, Yang, Zhenyu, Zhang, Zhaojing, Wang, Zitong, Wang, Bowen, Wu, Mingzhui. A Multi-Layer Intrusion Detection System for SOME/IP-Based In-Vehicle Network. Sensors: 28 April 2023, pp. 5-6

**For contacts:**

Eng. Ivan Orfey Ivanov, PhD student  
Computer Science and Engineering Department  
Technical University of Varna  
E-mail: iivaniivanov98@gmail.com



# АНАЛИЗ НА СЪВРЕМЕННИ ИНТЕЛИГЕНТНИ СИСТЕМИ ЗА МОНИТОРИНГ НА СЕЛСКОСТОПАНСКИ И ПИТОМНИ ЖИВОТНИ

Димитър В. Аврамов

**Резюме:** През последните години проследяването на здравето на животните с различни видове интелигентни системи се развива все повече. В статията е направен сравнителен анализ на такъв вид системи като са обхванати селскостопански животни и домашни любимци. Показателите и факторите, които се следят, са температура, кръвно налягане и др. Разгледаните системи са сравнени по различни показатели – микроконтролер, софтуер, технология, сензори.

**Ключови думи:** интелигентни системи, мониторинг на селскостопански животни, мониторинг на питомни животни, интернет на обектите

## Analysis of modern intelligent systems for monitoring agricultural animals and pets

Dimitar V. Avramov

**Abstract:** In recent years, animal health monitoring, with various types of intelligent systems, has been developing more and more. In the article, a comparative analysis of such systems was made, including farm animals and pets. The indicators and factors that are monitored are temperature, blood pressure, etc. The considered systems are compared according to various indicators - microcontroller, software, technology, sensors.

**Keywords:** intelligent systems, agricultural animals monitoring, pet monitoring, internet of things

### 1. Увод

Преди да бъдат описани различните видове интелигентни системи, е добре да се припомним как фермерите са наблюдавали добитъка си в миналото. Най-често, в ежедневната си работа, това се е случвало със стандартен аудио-визуален контакт. Като проблем, който може да се появи е, че този тип наблюдение няма как да не се прекъсне за 24-те часа в денонощието.

В наше време животновъдството се променя с бързи темпове. Голяма част от учените и фермерите прогнозира, че през следващите 15 години търсенето на месо ще нарасне с около 40%. За да се реши този, както и други проблеми, са необходими нови подходи и технологии за увеличаване на производителността. Всички тези решения трябва да подобрят грижата за здравето на животните и също така да са щадящи към околната среда [1].

В близкото бъдеще все по-голямо значение за животновъдството ще заемат място и модерните технологии за мониторинг на животински видове. Тези решения ще са от голяма ползва за изхранването на населението, както и за следенето на жизненоважни процеси при животните.

Интелигентните системи се използват за наблюдение, предсказване и анализ. Тези системи помагат за справянето с проблеми като:

- повишаване на качеството и количеството на селскостопанските животни;
- следене на миграцията и болестите по дивите животни;

- получаване на релевантна информация за пчелните семейства от разстояние, без да се отварят кошерите и да се прекъсва работния им процес;
- влиянието на разнообразни отпадъци на животните, които живеят във водата.

Затова в днешно време се използват различни технологии, като една от тях е IoT (интернет на обектите) [2].

Интернет на обектите (IoT) е технология, която най-често се свързва с изграждане на интелигентни домове. Освен това тази технология все повече навлиза в различни сектори, като един от тях е животновъдството. Също така към технологията се разработват различни сензори, предаватели и алгоритми за анализ на получените данни, приспособени към особеностите на дейностите и операциите, извършвани в различните отрасли.

При различните видове животни IoT технологията се развива в няколко направления:

- следене на храненията на животните;
- следене на движението на животните, когато са пуснати на паша;
- следене на поведението на животните:
  - при среща на животни от един вид;
  - при среща на животни от различен вид;
  - при среща на животни от различен пол;
- следене на различни жизнени показатели за превенция на здравословни проблеми.

## 2. Изложение

Чрез различните интелигентни системи могат да се наблюдават всякакви животински видове. Може да се следят диви животни, питомни животни, селскостопански животни, както и водни обитатели.

В зависимост от вида на животинския вид, който ще се следи, системите могат да бъдат разделени по следния начин:

- система, която да се закрепва върху добитъка;
- система, която да се постави на определено място – дърво, храст, стълб;
- система, която да се постави на хранилка или поилка;
- система, която да се постави в сграда;
- система, според вида на животното.

Според вида на изследваното животно, системите се делят на:

- интелигентни системи за мониторинг на селскостопански животни;
- интелигентни системи за мониторинг на питомни животни;
- интелигентни системи за мониторинг на пчели;
- др.

### 2.1. Интелигентни системи за мониторинг на селскостопански животни

Интелигентното животновъдство е една от областите, които биха могли да извлекат голяма полза от използването на съвременни технологии. Тя включва широко взаимодействие и съжителство между хора и животни и често се нарича също прецизно животновъдство (Precision Livestock Farming). Понастоящем интелигентното отглеждане на животни отговаря на основните нужди на фермерите, като например подпомагане на стопаните да автоматизират процеса - например, със системи за контрол на храненето.

Животните, които спадат към тази категория, са много, като част от тях са: крави; свине; домашни птици; кози; овце.

### 2.1.1. Интелигентни системи за наблюдение на млекодайки животни

При говедата, както и при други млекодайки животни, по-доброто им отглеждане довежда до по-добро противопоставяне на различни болести. От предишен опит е открито, че физиологичните и поведенческите резултати от изследвания на говеда, намиращи се в стопанството, предоставя информация за здравето им по време на паша. Пример за това са движенията на главата и скоростта, с която животното си движи главата. Тези показатели могат да помогнат както за здравето на животното, така и за качеството на млякото. Всяко движение, което се отличава от стандартното за добитъка по време на паша, може да привлече вниманието на стопанина. Каквито и промени да се забележат в поведението, ясно подсказват за появата на проблеми. Следователно могат да се използват като входни данни за система за мониторинг. Знаейки позицията, която са заели – легнало състояние или са изправени и ядат, има важна роля в производството на мляко. Системата, която е представена в [3], следи за няколко фактора, които са:

- телесна температура;
- гниене на копитата;
- болести, които постепенно отслабват животното;
- паразити;
- кучота.

Мобилните WSN (безжични сензорни мрежи) позволяват непрекъснато наблюдение на животните с по-висока устойчивост спрямо стандартното наблюдение. На нашите GPS системата се дава възможност да се отчете продължителността на пашата. От литературата се вижда, че нашите GPS системи са приложени успешно от няколко изследователи [5, 6].

Системата за мониторинг на селскостопански животни [7] се състои от две подсистеми - за животните и за околната среда. И двете подсистеми изпращат отчетените данни на фермерски контролер. Той (фермерският контролер) синхронизира данните чрез облака, който предоставя връзката с мобилното устройство на стопанина.

И тук, както при по-горната система, се наблюдават различни показатели като: температура; кръвно налягане; хранене; промени в поведението; прием на течности; социално поведение.

Хардуерната част на системата се състои от два броя микрокомпютри Raspberry Pi, които са за двете подсистеми, сензори за температура, влажност, жиромер, акселерометър, барометър, LED матрица, камера и джойстик за управление. Софтуерната реализация е базирана на езиките Python и Java, като Python е използван за предаване на данните от микрокомпютрите, а на Java е написано мобилното приложение. Мобилното приложение предоставя различни възможности, като фермерът може да избере кои характеристики да следи, както и кои устройства да управлява – вентилация, затваряне и отваряне на врати и др. Базата от данни, която е използвана, е MySQL, а комуникацията между отделните компоненти на системата е реализирана чрез TCP връзка. Сигурността е базирана на TLS протокол.

### 2.1.2. Интелигентни системи за наблюдение на домашни птици

През последните години отглеждането и консумацията на домашни птици са нараснали в цял свят [8] и това се дължи на търсенето на високо съдържание на протеини, ниско съдържание на натрий и холестерол.

Средата в кокошарника е много важен фактор при отглеждането на домашните птици, защото качеството на месото и яйцата до голяма степен зависят от условията, които са осигурени. Това от своя страна налага постоянно следене на температурата, качеството на



въздуха, проветряването на помещението, влажността, концентрацията на различни видове химични елементи или съединения (газове) – въглероден диоксид, амоняк и др.

PMS (Poultry Monitoring System) [9] използва ESP32 микроконтролер, DHT11 сензор за температура и влажност, PIR сензор за движение и MQ135 сензор за газ.

За следенето на качеството на въздуха, сензорът, който е използван, предоставя информация за нивата на: NH<sub>3</sub>; NO<sub>x</sub>; алкохол; бензен; дим; CO<sub>2</sub>.

PIR сензорът може да засича животно или човек, които се намират в близост до кокошарника. Пилетата са предразположени за нападение от хищници като котки, кучета, лисици или да бъдат откраднати от човек. Това обосновава нуждата от използването на сензора.

DHT11 се използва за засичане на температурата и влажността в птицефермата. Идеалната температура за птиците е около 20°C или 70°F, а влажността – между 60% и 80%. Датчикът по подразбиране е настроен, което улеснява работата с него.

Алгоритъмът за работата на системата се състои от следните стъпки:

- Инициализация на микроконтролера и сензорите;
- Проверка при свързване към безжичната мрежа;
- Прочитане на стойностите от сензорите;
- Изпращане на информацията от сензорите към Blynk приложение.

Софтуерът е написан на езика C++. Използват се различни програмни интерфейси (API), които предоставят допълнителни възможности на ESP32 микроконтролера, а за извеждане на данните от сензорите е използвано Blynk.

При системата [10] се дава възможност на фермера да наблюдава птицефермата и да контролира дейността ѝ. Системата е комбинация от безжични сензори и мобилна система, която може да се управлява и наблюдава. Параметри на околната среда като температура, интензитет на светлината и газ (амоняк) също се наблюдават и контролират автоматично [11]. Чрез Интернет се свързват устройствата, като им се осигуряват разнообразни начини за комуникация. Интелигентните системи могат да спестят пари, време и труд.

Тази система намалява човешкия труд – не се налага човек да наблюдава постоянно има ли храна или не. Системата включва две главни секции - първата е за подаване на храната в контейнерите, а втората е за контролиране на температурен сензор, който да следи за свежестта на храната за пилетата [12]. Също така се подобряват следните условия:

- климата на домашните птици;
- намаляват се разходите за работна ръка;
- спестява се храна;
- храненията на пилета са навреме;
- избягва се замърсена храна от насекоми.

Системата използва различни сензори и open-source софтуер. Може да отчита температура, влажност, интензитет на светлина и качество на въздуха. Проектирана е така, че фокусът да пада върху използването на IoT, нискобюджетен хардуер и свободен софтуер. Тя спестява време, намалява труда от гледна точка на работна ръка и повишава продукцията.

Системата изпълнява следните задачи - първо, потребителят задава команда като включване на системата или команда за някакво движение чрез приложение, което използва Bluetooth технология – с други думи, на Bluetooth модула се изпраща информация, която се предава на Arduino UNO. Второ, поливането с вода – ако температурата се повиши на границата, която е зададена, на потребителя се изпраща известие, а той вече може да контролира моторите, които отговарят за помпата с водата. Третата задача е използването на вила за премахването на отпадъците, да се намали създаването на газове.

Хардуерът на системата се състои от Arduino UNO, DHT22, Bluetooth модул, сензор за разстояние, драйвер за контролиране на мотори.

Алгоритъмът се разделя на няколко части, в зависимост от това, което трябва да се управлява:

- пълненето на хранилката;
- контролиране на температурата;
- намаляване на газовете;
- засичане на препятствия.

Математическият модел ( $S = (I, O, F, SP, G)$ ), който се използва, се основава на множество от входни данни, множество от изходни устройства за управление, пълненето на хранилката и контролирането на газовете.

### 2.1.3. Системи за наблюдение на животни, отглеждани за месо

В системата [13] е проектирано IoT приложение за поддръжка на малки свинеферми. Популацията на това проучване е взета от SMCE (Small and Micro Community Enterprise) ферми в провинция Nakhon Si Thammarat, Тайланд. Построената свинеферма е с повдигнат покрив за вентилация, в която се отглеждат 120 прасета, разделени в 4 отделения/бокса. Ежедневното задължение на фермера е да изсипва храна във въртящ се механичен бункер. Вечерта фермерът наблюдава дали прасетата имат достатъчно храна. Освен това фермерът трябва да записва количеството храна, което прасетата изяждат на ден. Също така, едно от главните задължения е наблюдението на животните. В горещ ден фермерът пръска вода в клетката, като целта е да се избегне топлинен удар на прасетата. В студените дни фермерът включва осветлението/отопление, за да им осигури топлина.

Проектираната IoT система е за наблюдение на температура, влажност, ниво на храна и контрол на храненето. Използваният хардуер и софтуер е с отворен код и позволява безжичен достъп [14]. Разработването на системата е обяснено в три фази, които са хардуерен дизайн, софтуерен дизайн и тестване.

За изграждането на IoT платформа на локален сървър е използвана NoSQL база от данни, събрани от IoT устройство, като е използван JSON и InfluxDB. Базата от данни позволява мащабируемост, висока достъпност, бърз запис и по-бързо четене. Node-RED е инструмент за програмиране, който използва браузър (уебсайт или API) редактор, което го прави лесен за свързване на разнообразни възли.

MQTT протоколът се използва за прехвърляне на данни от микроконтролер (MQTT клиент) към Node-RED сървър. Системата представя „Съхранение на данни“, „Постоянна визуализация на данни“ и „Мониторинг в реално време“. Разработеното решение обхваща връзките между сървър и база от данни.

Използван е следният хардуер: WEMOS D1 Mini (ESP8266) с вграден 32-битов RISC микроконтролер, работещ на 80MHz. Разполага с Wi-Fi приемопредавател, 64KB RAM за инструкции, 96KB RAM за данни и 4MB флаш памет. DHT11 е сензор, който отчита температурата и влажността. Работи в следния диапазон: температура – 0 - 60°C, влажност - 20% ~90%. В системата намират място и сензор за разстояние – HCSR04, и релета, с които се управляват вентилатори, редукторни мотори и осветлението.

Представени са два алгоритъма: един за следене на температурата и влажността и един за хранилката.

Тестването използва http заявка, която изпраща и получава данните като JSON от единият микроконтролер. Получената информация се изпраща към базата от данни, откъдето потребителите могат да следят показателите. Данните се извеждат, като се използва NodeRED Dashboard package.

Информацията се анализира и визуализира с помощта на Grafana, а за съхранение се използва базата от данни InfluxDB, която е инсталирана на сървъра.

В китайското животновъдство овцевъдството е индустрията, която заема второ място след млечната промишленост. През 2017 г. има 302,3 милиона живи овце за продажба, което е увеличение от 1% спрямо 2016 г. През 2017 година Китай е произвел 4,7 милиона тона овнешко месо, което е увеличение от 2,3% спрямо предходната година. Броят на живите овце в Китай достигна най-високото си ниво през 2015 г., достигайки 311,7 милиона [15].

Като един от най-важните източници на протеин в ежедневието на хората, качеството и безопасността на овнешкото месо е тема, която все по-широко се застъпва [16].

Транспортирането на живи овце е от ключово значение във веригата за доставка за месо. Най-важната задача е да се избегне смъртта на овцете по време на транспортирането и щетите, причинени от прекомерния стрес при овцете. Въпреки това има малко проучвания относно това как да се намали реакцията на стрес и промените в качеството на овнешкото месо по време на транспортиране. Понастоящем транспортирането на живи овце се извършва главно в условията на големи и претъпкани камиони. Транспортната среда лесно се влошава поради влиянието на дишането и метаболизма на животните. Ако не се почисти навреме, това ще предизвика симптоми на стрес при овцете и ще причини заболяване, като може да се стигне дори и до смърт. Ето защо, безопасното транспортиране на овче месо винаги е бил основен проблем на пазара.

Технологията, която използва преносими сензори, е нова, но използването ѝ при домашните животни е от голямо значение [17-21].

Овцете са много активни и предпазливи по време на паша, като не харесват поставянето на чужди тела, като електроди. Следователно системата за мониторинг на стреса трябва да бъде не само устойчива, но и стабилна. Предизвикателна задача е надеждното записване на сърдечната честота и температурата на кожата на овцете, когато не се движат. Първата стъпка е да се подобри хардуерът на фотоплетизмограмата за по-ефективно интегриране и разработване на по-миниатюрно, точно и надеждно оборудване за наблюдение на PPG извън общата лабораторна и ветеринарна среда, особено по време на динамично транспортиране. Необходимо е да се приложи безжична система за мониторинг от типа „главно устройство – подчинено устройство“ чрез интегриране на Bluetooth и IoT технологиите, за да се намали дискомфортът на овцете.

Преносимата система за мониторинг на стреса (WSMS) включва два взаимосвързани компонента: главен модул, фиксиран в регулируема еластична лента, която е монтирана на гърба на овцата зад двата предни крака около гръдния кош, и подчинен модул, окачен на около шията [22].

WSMS (Wearable Stress Monitoring System) включва два вида данни:

- данни за наблюдение на биологични признаци;
- данни за наблюдение на околната среда.

Сензорите за сърдечен ритъм и инфрачервени температурни сензори в подчиненото устройство измерват данните за наблюдение на биологичните признаци, включително сърдечната честота и температурата на кожата. Съответните сензори в главното устройство изпращат данни от наблюдението на транспортната среда, включително глобална система за позициониране (GPS), триосов акселерометър, температура на околната среда и влажност.

Сензорите в подчинения модул включват сензор за сърдечен ритъм и сензор за температура на кожата, които се фокусират върху физиологичната информация на овцете.

WSMS включва главно устройство и подчинено устройство. Главният модул е дефиниран като устройство, което има възможности за обработка, получаване, обработка и запис на данни. Подчиненият модул е представен като устройство, което отговаря за събирането на и изпращането на данни до главното устройство. Причината, поради която е използван начинът на предаване на данните главен-подчинен, е да се намали колкото е възможно размерът на устройството за събиране на данни.

### 2.1.4. Технологични, софтуерни и хардуерни решения

В таблица 1 е представено сравнение по различни показатели на описаните интелигентни системи за мониторинг на селскостопански животни.

**Таблица 1.** Сравнение на описаните интелигентни системи за селскостопански животни

Система Характеристики	Smart Farm System	Cattle health monitoring system using wireless sensor network	Poultry Monitoring System	IOT Based Smart Poultry Farm	Smart farm prototype for pig farming	Wearable Stress Monitoring System
Използван контролер	Raspberry Pi	-	ESP32	Arduino UNO	ESP8266	ATmega328P
Технология	IoT	WSN	IoT	IoT	IoT	IoT
Сензор за температура (телесна)	✓	✓	-	-	-	✓
Сензор за температура (помещение)	✓	-	✓	✓	✓	-
Сензор за влажност	✓	-	✓	✓	✓	✓
Сензор за кръвно налягане	-	-	-	-	-	✓
Сензор за сърдечен ритъм	-	✓	-	-	-	✓
Сензор за положение	✓	-	-	-	-	-
Сензор за разстояние	-	-	-	-	✓	-
Сензор за движение	-	-	✓	-	-	✓
Сензор за газ	-	-	✓	✓	-	-
Сензор за отчитане на интензитета на светлина	-	-	✓	-	-	-
GPS	-	✓	-	-	-	✓
Камера за наблюдение	✓	-	-	-	✓	-
Мобилно приложение	✓	-	✓	✓	-	-
Desktop приложение	✓	-	-	-	-	-
Web приложение	-	-	-	-	✓	-
База от данни	MySQL	-	-	-	InfluxDB	-

### 2.2. Интелигентни системи за мониторинг на питомни животни (домашни любимци)

Когато става дума за мониторинг на животни, една от насоките е към питомните животни. Те, както и другите животни, не могат да говорят, за да си кажат проблемите или болестите. Затова се разработват и прилагат различни интелигентни системи, които да показват температурата, кръвно налягане, сърдечен ритъм и др.

Връзката между хората и животните има много предимства за човешкото благосъстояние [65]. Различни проучвания показват, че грижата за животните може да подобри качеството на живот на индивидите в психологически и физически аспект [66].

### 2.2.1. Интелигентни системи за мониторинг на кучета

Разпространението на домакинствата с кучета и други домашни любимци се е увеличило през последните години [29]. Тези животни, известни като животни-компаньони, съжителстват с хората, като предлагат емоционална подкрепа, помагат на собствениците си да поддържат физическа и умствена дейност и изграждат дълбоки връзки [51-56]. Тази тенденция води до нарастващ интерес към здравето на питомните животни, успоредно с нарастващия им брой в домовете [30]. В здравеопазването появата на различни технологии позволява на хората да обсъждат и наблюдават здравословното си състояние с помощта на мониторинг в реално време и диагностични системи с изкуствен интелект [31]. Общуването при кучетата е предимно чрез жестове и поведение, което представлява предизвикателство за количествена оценка на здравословния статус [32].

Например, поведението на кучето при миризма на земя може да показва игривост, стрес или болест. Бързата загуба на тегло може да означава диабет или бъбречно заболяване, докато наддаването на тегло може да означава преяждане или липса на упражнения. Честото чесане в областта на очите може да е симптом на кератит или конюнктивит. Въпреки това, собствениците без ветеринарни познания може да се затруднят да разпознаят тези специфични поведенчески или физически промени [33]. Животозастрашаващи спешни случаи могат да възникнат от нелекувани възпалителни кожни заболявания [34]. Следователно, идентифицирането на специфични, необичайни поведения е от решаващо значение за оценка на здравето на кучето и осигуряване на подходящо лечение [35].

Многобройни проучвания са изследвали прогнозирането на болести по кучетата чрез анализиране на необичайни поведенчески модели като се използват преносими сензори [36-38]. Чрез прикрепване на сензор по начин, който не възпрепятства ежедневните дейности на домашния любимец, наблюдението става по-достъпно, като се улеснява откриването на необичайни модели. Сензори за активност, традиционно използвани за наблюдение на човешкото здраве и широко прилагани в области като рехабилитация, здравеопазване и здравни показатели [39], сега се внедряват в системи, които следят здравето на кучетата. Въпреки че множество проучвания оценяват състоянието на човешкото здраве [40], разработването на системи, свързани със здравето на кучетата, е голямо предизвикателство поради променливи като порода, възраст и тегло, които са трудни за количествено определяне с прости модели на данни.

Системата [41] въвежда усъвършенстван сензор за активност, включен в многостранен алгоритъм, за измерване на нивата на активност въз основа на специфичното поведение на кучетата. Целта на системата е разработването на метод за извеждане на здравословното състояние на кучетата.

Проучването е реализирано в рамките на девет месеца и е проведено върху тридесет кучета. За да се гарантира точното разбиране на съществуващите заболявания и здравословни състояния преди началото на експеримента, всички участващи кучета са преминали преглед във ветеринарна болница. За аналитични цели кучетата са категоризирани въз основа на теглото в малки, средни и големи групи и допълнително класифицирани по възраст.

Данните в системата се събират от сензора за активност, а също така се използва и камера, която записва поведението на кучето. Поведенията са категоризирани в статични и динамични типове. Динамичното поведение при кучетата включва дейности като бягане, скачане, хранене и преглъщане. Обратно, статичните поведения са тези, при които позицията на кучето остава постоянна, като седене, лежане и сън [42]. Измерванията на активността, получени от изображенията и данните от ускорението и жироскопа, преминават процес на филтриране. Критериите за филтриране се установяват чрез корелиране на стойностите на

данните от сензора с действителните изображения, като по този начин се повишава точността и пълнотата на данните.

Алгоритъмът за обучение използва алгоритъм за асоциативна памет. Този алгоритъм филтрира подобни модели и използва асоциативни техники за съпоставяне на идентични данни след потвърждаване на поведенческия анализ на кучетата [43]. Сензорът за поведение на кучетата идентифицира четири ключови поведения, като те са установени за референтни точки за всяка порода. За поведението при чесане и облизване е използвана четиристепенна класификационна система: случайно, средно, често и тежко. Следователно минималното драскане и близане, открити от сензора, предполагат здравословно състояние без заболяване, докато прекомерните прояви на тези поведения могат да показват кожни заболявания [44]. Нивото на гълтателна активност се определя количествено чрез броя на гълтанятия на час. Брой, по-нисък от средното ниво, измерен от сензора, може да предполага храносмилателни заболявания или нарушения на хранопровода, докато по-високият брой може да показва прекомерен прием на вода, потенциално симптоматичен за хронично бъбречно заболяване [45].

Здравният резултат е количествена метрика, която е разработена, за да улесни визуализацията на първоначалните резултати от анализа на активността. Този резултат отразява здравословното състояние на кучето и дава ясна индикация за потенциални промени въз основа на бъдещи данни [46]. Здравният резултат се изчислява по скала от 1 до 10. Резултат 1 означава ниско ниво на здраве и повишен риск от заболяване, докато по-високите резултати означават по-добро здраве. AI-предсказан здравен резултат, вариращ в диапазона от 1 до 5, предполага съществуващи здравословни проблеми, които изискват медицинска помощ. Резултатите между 6 и 10 са показателни за добро здравословно състояние. Валидността на тези оценки е потвърдена от трима ветеринарни лекари [47].

В системата [48] е показан прототип на система, която може да се постави върху животното и показва следните стойности: температура; сърдечен ритъм; дихателна дейност; кръвно налягане; ЕКГ.

Системата събира изброените по-горе стойности от няколко сензора, след което ги изпраща чрез IoT. Ветеринарният лекар има достъп до данните и поставя диагноза на животното. Всички сензори се включват в Arduino UNO, а резултатите се показват на LCD дисплей.

Сензорите, които се използват в системата, са: сензор за температура; сензор за дихателна дейност; сензор за кръвно налягане; сензор за ЕКГ и сърдечен ритъм.

Arduino UNO е развойна платка, която се състои от микроконтролер 328p, съединители за хранване, GPIO изводи и др. Платката може да се програмира от Arduino IDE чрез USB-to-USB кабел.

Сензорът за температура (DS18B20) работи в диапазона от -55 до +125°C, а нормалната ректална температура на кучето е от 37.9 до 39.9°C. Всяка стойност, която не попада в този диапазон, е предпоставка за завеждане на животното при ветеринар.

Сензорът за дихателна дейност се състои от сензор за налягане и колан, който трябва да се закачи за гърдите на животното. Когато домашният любимец си поема въздух, белите дробове се разширяват, което избутва гръдния кош. По този начин се отчита разликата в налягането и се оценява дихателната дейност.

Сензорът за кръвно налягане отчита кръвното налягане като използва осцилометричен метод. Този метод се прилага при електронните апарати, като уредът регистрира пулсациите на налягането на въздуха, които са възникнали в маншета при преминаването на кръвта през участъка, който е бил притиснат.

Последният модул, който се използва в системата, е ЕКГ модулът. Той отчита пулса, ритъма и проводимостта на сърцето [50]. Използват се три Ag-AgCl електрода, като два от тях се поставят на предните крайници и един на левия заден крайник. Добавя се специален

гел, който поставя между кожата и електродите, като така се намалява грешната информация [49].

### 2.2.2. Интелигентни системи за мониторинг на котки

Първите четири седмици за новородените котета са най-критични, защото нямат възможността да контролират температурата на тялото си и да се приспособяват към температурата на околната среда. Смъртността през първите 9 седмици е между 6 и 20% [57]. Терморегулацията е способност при животните, която контролира температурата на тялото.

В случаите, когато котето е без майка или е много малко и не може да контролира температурата си, се налага използването на инкубатор. Важно е животинчето да бъде на топло, но трябва да се внимава с топлината, за да не се дехидратира, да не се шокира тялото с прекалена топлина, защото може да се стигне до фатален край.

Наблюденията във и извън инкубатора се правят ръчно, като представляват включването и изключването на светлина, чрез която се контролира температурата. За целта е изработен метод, който включва система, базирана на IoT. Мониторингът е много важен както за специалистите по здравеопазване на животните, така и за любителите на котки, когато наблюдават температурата и влажността в инкубатора чрез интернет, което позволява дейността да се извършва по всяко време и навсякъде от смартфон чрез приложението Blynk [58].

Системата [59] показва разработването на инкубатор, в който е внедрена технологията IoT. Няколкото етапа, които са извършени при създаването ѝ, се състоят от проектиране и производство на инкубаторната плоча, тестване и приложение. Проектирането и производството на инкубаторната плоча е началният етап, при който инкубаторът се изработва без никакви допълнителни функции. След като инкубаторът е направен, се прилага Интернет на обектите (IoT) към него [60]. Прилагането на тези компоненти съвпада с разработването на приложен софтуер, който е интегриран с хардуерни компоненти, които се прилагат към инкубатора, докато отделните компоненти се свържат, за да могат да работят онлайн. С прилагането на онлайн базиран приложен софтуер наблюденията могат да се извършват дистанционно чрез мобилни телефони. Показателите, които се наблюдават, са температурата и влажността, защото те се явяват критични точки при използването на инкубатори за отглеждането на новородените котета.

При системата на Rung-Ching Chen [61] се следи за промяна на поведението на домашния любимец всеки ден и се изпраща известие, в случай че е засечена такава.

Тя се състои от следните компоненти:

- Raspberry Pi е микрокомпютър, който позволява инсталирането на Linux ОС, и е проектиран за обучение. През последните години е подобряван постоянно, като тук се включва и физическия му размер. В системата е използван микрокомпютър и камера.
- Deep Learning е подраздел на машинното обучение. Този тип обучение е помогнало на учените да постигнат добри резултати както в разпознаването на предмети или лица, така и в биомедицината. В системата на [62] е използван самообучаващ се CNN алгоритъм и SVM модел за разпознаване на лицата на 10 прасета. В крайна сметка SVM постига резултат от 91%, а CNN алгоритъмът постига резултат от 96.7%.
- YOLO за първи път е споменат през 2015 година от Joseph Redmon. Името идва от първите букви – You Only Look Once [63]. Това е система, която засича обекти в реално време, а в основата ѝ стои SNN (single neural network). За разлика от други алгоритми, YOLO разделя изображението на множество сегменти и прогнозира координатите на ограничаващата кутия на всяка единица. YOLO предоставя добра

скорост и точност при засичане на обекти и поради тази причина е предпочитан избор от много учени [64].

Системата използва YOLOv3-Tiny модел за обучение на събраните изображения на котки. Множество Deep Learning алгоритми, комбинирани с Raspberry Pi, позволяват разпознаването по снимки да бъде още по-ефективно. В основата стои камера, свързана към Raspberry Pi. След като се направят снимки, се включва моделът за разпознаване по снимки, който е установен на микрокомпютъра, заедно с платформа за мрежова услуга IFTTT. IFTTT е облачна платформа за хора, които трябва да проектират автоматизирани интелигентни услуги (домове). За преглеждането на изображенията отдалечено, е използван VNC (Virtual network computing). VNC се състои от клиент и сървър. Сървърът е инсталиран на Raspberry Pi, а клиентът – на компютъра или мобилното устройство.

Алгоритъмът на работа на системата е следният:

1. Изображение в реално време, което се създава от камерата.
2. Направените изображения се изпращат в реално време към монитор или мобилно устройство, като се използва VNC (клиент-сървър).
3. Ако микрокомпютърът засече промяна в поведението, то на мобилното устройство се изпраща изображението с промяна в поведението, чрез IFTTT .
4. За да се установи промяна в поведението на домашния любимец, получените снимки се предават на YOLOv3-Tiny модела.

Системата за наблюдение улавя поведението на домашния любимец при сън, хранене, сядане и ходене и собственикът може да наблюдава промяната в ежедневните им дейности. Така системата за мониторинг изпраща известие до собственика, когато домашният любимец е до тоалетна за повече от 30 секунди и търси в кофата за отпадъци. По този начин собственикът не е претоварен със съобщения, но в същото време може да наблюдава последващите дейности на домашния любимец.

За обучението на YOLO са използвани 246 снимки, на които котето върви; 180 снимки, на които яде; 232 снимки, на които спи; 210 снимки, на които е седнало; 260 снимки, на които търси в кофата с отпадъци; 282 снимки, на които отива до тоалетна.

Системата [67] дава възможност на потребителите да контролират храненето на домашните си любимци и да проследяват състоянието им.

Хардуерната част на системата се състои от ESP32 микроконтролер, DHT22 сензор за температура и влажност, серво мотор, клетка за тегло и ESP32-Cam. С използването на Интернет се позволява стойностите да се съхраняват в база от данни.

Потребителите могат да използват всички възможности, но след регистрация. След успешна регистрация, на ползвателите на web приложението се дава възможност с какво да продължат – с контролирането на храната или с проследяването на показателите на питомните животни.

Софтуерната част е представена от web приложение, което позволява контролирането на хранилката и проследяването на показателите, както и възможността да се гледа животното в реално време.

### **2.2.3. Технологични, софтуерни и хардуерни решения**

В таблица 2 е представено сравнение по различни показатели на описаните интелигентни системи за мониторинг на домашни животни. Единственият показател, който се покрива от всички системи, е използваната технология Интернет на обектите (IoT).



**Таблица 2.** Сравнение на описаните интелигентни системи за питомни (домашни) животни

Характеристики \ Система	Wearable Alert for Animal Health Monitoring System	Temperature and Humidity Monitoring System Environmental	Dog Health Score Using an Artificial Intelligence Disease Prediction Algorithm	Monitoring the behaviours of pet cat	Control and Monitoring Systems in Internet of Things (IoT) Based Cat Cage
Използван контролер	Arduino UNO (328p)	ESP8266	-	Raspberry Pi	ESP32
Технология	IoT	IoT	IoT	IoT	IoT
Алгоритъм/Модел	-	-	✓	YOLOv3	-
Сензор за температура (телесна)	✓	✓	-	-	-
Сензор за температура (помещение)	-	✓	-	✓	✓
Сензор за влажност	-	✓	-	✓	✓
Сензор за кръвно налягане	✓	-	-	-	✓
Сензор за сърдечен ритъм	✓	✓	-	-	✓
Сензор за положение	-	-	✓	-	-
Сензор за движение	-	-	✓	-	✓
Камера за наблюдение	-	-	-	✓	✓
Мобилно приложение	-	✓	✓	✓	-
Desktop приложение	-	-	-	✓	-
Web приложение	✓	-	-	-	✓
База от данни	-	-	-	-	✓

### 2.3. Основни показатели, които се следят от интелигентните системи

Автоматичните системи за мониторинг и събиране на данни позволяват анализирането на вече събраните данни, които включват физиологичните реакции, движението, поведението на животните и засичането на болести. Най-често при животните, независимо от вида – селскостопански или домашни, се проследяват, във времето, следните показатели и фактори [23-28]:

- Телесна температура – Стресът при животните най-често се оценява с измерването на телесната температура. Ветеринарите използват ректални термометри за тази цел от години [4]. Отчитането на тази характеристика може да се получи от няколко места. Нормалната температура на животните е 38.6°C. Като цяло границите могат да варират от 37.8°C до 40.0°C;
- Гниене на копитата – получава се, когато почвата е кална и влажна/мокра [3];
- Болести, които постепенно отслабват животното – това са болести като болестта на Джон и туберкулоза, които са следствие на една и съща бактерия [3];
- Production diseases – Това са болести, които са свързани с метаболизма, генетиката или инфекция, примерно, задържане на плацента. Най-големите загуби, както на животни, така и на мляко са свързани с този тип болести [3];

- Паразити – двата вида паразити, които оказват влияние, са вътрешни и външни. Вътрешните влияят на теллото на животното и количеството на млякото. Външните паразити като кърлежи и мухи повлияват на здравето им [3];
- Куцота – невъзможността на добитъка да се движи свободно [3];
- Кръвно налягане – показател, който дава информация за това дали животното има признаци за увредени органи [7, 48];
- Промени в поведението [7];
- Прием на течности [7];
- Социално поведение [7];
- Сърдечен ритъм [22, 48];
- Дихателна дейност [48].

В резултат на направения аналитичен обзор могат да бъдат направени следните изводи за системите от този вид:

- В голяма част от системите липсват бази от данни;
- В голяма част от системите не се отчитат показатели като кръвно налягане и телесна температура;
- Промяната на поведение на животното подсказва за наличието на проблем;
- В голяма част от системите информацията не се извежда и в desktop или уеб приложение;
- В повечето случаи сензорите не са обединени в един модул или в една допълнителна платка;
- В малка част от системите се използва изкуствен интелект;
- В малка част от системите се дава право на собственика на животните да контролира някакъв вид устройство.

### 3. Заключение

В изложението са представени различни системи за наблюдение на животни. Всяка една от тези системи има своите предимства и недостатъци.

Предимствата се изразяват в това, че повечето системи имат или могат да използват готово мобилно приложение. Също така цената на системите е сравнително ниска - и като отделни елементи (сензори и контролер), и като краен продукт (хардуер и софтуер).

Недостатъците на предложените системи се появяват най-вече в многото отделни сензори и липсата на база от данни (за голяма част от системите). Тъй като датчиците не са обединени, примерно, в един модул, а се използват няколко, така те заемат по-голяма площ. Другият недостатък е липсата на база за данните, които се получават – те се извеждат, но не се съхраняват, за да може да се направи сравнение, както и прогноза, при нужда.

Тенденцията при разработката на такъв тип системи е внедряването на изкуствен интелект, за да може с негова помощ да се откриват по-бързо и по-лесно болестни и нетипични поведения.

### Литература

- [1]. [https://agrobio.elmedia.net/bg/2019-3/editorials/iot-в-животновъдството\\_02006.html](https://agrobio.elmedia.net/bg/2019-3/editorials/iot-в-животновъдството_02006.html).
- [2]. <https://bg.profiland.net/articles/smart-sistemi-za-kontrol-i-upravlenie-na-zhivotnovdstvoto>.
- [3]. Bhisham Sharma, Deepika Koundal, “Cattle health monitoring system using wireless sensor network: a survey from innovation perspective”, The Institution Engineering and Technology, 16th March 2018.

- [4]. Lukonge, A.B., Kajjage, D.S., Sinde, R.S.: ‘Review of cattle monitoring system using wireless network’, *International Journal of Engineering and Computer Science*, 2014, 3, (5), pp. 5819–5822.
- [5]. García-Hernández, C.F., Iburguengoytia-Gonzalez, P.H., García-Hernández, J., et al.: ‘Wireless sensor networks and applications: a survey’, *Int. J. Comput. Sci. Netw. Secur.*, 2007, 7, (3), pp. 264–273.
- [6]. Huircán, J.I., Muñoz, C., Young, H., et al.: ‘ZigBee-based wireless sensor network localization for cattle monitoring in grazing fields’, *Comput. Electron. Agric.*, 2010, 74, (2), pp. 258–264.
- [7]. Marcel Caria, Jasmin Schudrowitz, Admela Jukan and Nicole Kemper, “Smart Farm Computing Systems for Animal Welfare Monitoring”, 2017, May 22- 26, 2017, Opatija, Croatia.
- [8]. Escobedo del Bosque et al., 2021, Arrieta & González, 2019.
- [9]. Stella Ifeoma Orakwue, Hamza Mohammed Ridha Al-Khafaji, Mays Zahid Chabuk, IoT Based Smart Monitoring System for Efficient Poultry Farming, *Webology*, Volume 19, Number 1, January, 2022.
- [10]. Shubham Mitkari, Ashwini Pingle, Yogita Sonawane, Sandip Walunj, Anand Shirsath, IOT Based Smart Poultry Farm, *International Research Journal of Engineering and Technology (IRJET)*, Volume: 06 Issue: 03 | Mar 2019.
- [11]. Archana M P1, Uma S K2, “Monitoring and controlling of poultry farm using IOT”, *International Journal of Innovative Research in Computer and Communicatio Engineering*, Vol. 6, Issue 4, April 2018.
- [12]. Zainal H. C. Soh1, Mohd H. Ismail1, “Development of automatic chicken feeder using Arduino Uno”, *IEEE*, Dec 2017.
- [13]. Pairot Sena, Boonnipa Kaiwman, “The Use of Internet of Things technology to develop a smart farm prototype for pig farming”, Faculty of Science and Technology, Nakhon Si Thammarat Rajabhat University, Nakhon Si Thammarat, 80280 Thailand.
- [14]. K.S.N.K. Marwat, Y. Mehmood, F. Ullah, A. Khan, S. Khan, S. Ahmed, D. Kwak, A. Nazir, Mobile Wi-Fi based Scheduling of Cyber-Physical Systems in Healthcare, *Electronics*. 9(2) (2020) 247.
- [15]. Sheng, Y.; Song, L. Agricultural production and food consumption in China: A long-term projection. *China Econ. Rev.* 2019, 53, 15–29.
- [16]. Zhou, D.; Yu, X.; Herzfeld, T. Dynamic food demand in urban China. *China Agric. Econ. Rev.* 2015, 7, 27–44., Hovhannisyan, V.; Mendis, S.; Bastian, C. An econometric analysis of demand for food quantity and quality in urban China. *Agric. Econ.* 2019, 50, 3–13.
- [17]. Neethirajan, S. Sensing and Bio-Sensing Research Recent advances in wearable sensors for animal health management. *Sens. Bio-Sens. Res.* 2017, 12, 15–29.
- [18]. Iomt, M.T.; Qureshi, F.; Krishnan, S. Wearable Hardware Design for the Internet of Medical Things (IoMT). *Sensors* 2018, 18, 3812.
- [19]. Yoon, H.; Xuan, X.; Jeong, S.; Park, J.Y. Biosensors and Bioelectronics Wearable, robust, non-enzymatic continuous glucose monitoring system and its in vivo investigation. *Biosens. Bioelectron.* 2018, 117, 267–275.
- [20]. Paper, E.S.A.C. The golden age of bio-logging: How animal-borne sensors are advancing the frontiers of ecology. *Ecology* 2015, 96, 1741–1753.
- [21]. Griffies, J.D.; Zutty, J.; Sarzen, M.; Soorholtz, S. Wearable sensor shown to specifically quantify pruritic behaviors in dogs. *BMC Vet. Res.* 2018, 14, 124.
- [22]. Yan Cui; Mengjie Zhang; Jun Li; Hailing Luo; Xiaoshuan Zhang; Zetian Fu; WSMS: Wearable Stress Monitoring System Based on IoT Multi-Sensor Platform for Living Sheep Transportation.

- [23]. Sendra S, Llarío F, Parra L, Lloret J (2013) Smart Wireless Sensor Network to Detect and Protect Sheep and Goats to Wolf Attacks. *Recent Advances in Communications and Networking Technology* 2(2):91-101.
- [24]. Nadimi ES, Jørgensen RN, Blanes-Vidal V, Christensen S (2012) Monitoring and classifying animal behavior using ZigBee-based mobile ad hoc wireless sensor networks and artificial neural networks. *Computers and Electronics in Agriculture* 82:44-54.
- [25]. Mackay JRD, Deag JM, Haskell MJ (2012) Establishing the extent of behavioural reactions in dairy cattle to a leg mounted activity monitor. *Applied Animal Behaviour Science* 139:35-41.
- [26]. Martins RFS, Paim T do P, Cardoso C de A, Dallago BSL, Melo CB de, Louvandini H, McManus C (2013) Mastitis detection in sheep by infrared thermography. *Research in Veterinary Science* 94(3):722-724.
- [27]. Mcmanus C, Tanure CB, Peripolli V, Seixas L, Fischer V, Gabbi AM, Menegassi SRO, Stumpf MT, Kolling GJ, Dias E (2016) Infrared thermography in animal production: An overview. *Computers and Electronics in Agriculture* 123:10-16.
- [28]. Alsaad, M, Syring C, Dietrich J, Doherr MG, Gujan T, Steiner A (2014) A field trial of infrared thermography as a non-invasive diagnostic tool for early detection of digital dermatitis in dairy cows. *The Veterinary Journal* 199:281-285.
- [29]. Aich, S.; Chakraborty, S.; Sim, J.S.; Jang, D.J.; Kim, H.C. The design of an automated system for the analysis of the activity and emotional patterns of dogs with wearable sensors using machine learning. *Appl. Sci.* 2019, 9, 4938.
- [30]. Christian, H.; Bauman, A.; Epping, J.N.; Levine, G.N.; McCormack, G.; Rhodes, R.E.; Richards, E.; Rock, M.; Westgarth, C. Encouraging dog walking for health promotion and disease prevention. *Am. J. Lifestyle Med.* 2018, 12, 233–243.
- [31]. Vassli, L.T.; Farshchian, B.A. Acceptance of health-related ICT among elderly people living in the community: A systematic review of qualitative evidence. *Int. J. Hum.-Comput. Interact.* 2018, 34, 99–116.
- [32]. Laroute, V.; Chetboul, V.; Roche, L.; Maurey, C.; Costes, G.; Pouchelon, J.-L.; De La Farge, F.; Boussof, M.; Lefebvre, H.P. Quantitative evaluation of renal function in healthy Beagle puppies and mature dogs. *Res. J. Vet. Sci.* 2005, 79, 161–167.
- [33]. Gruen, M.E.; Lascelles, B.D.X.; Colleran, E.; Gottlieb, A.; Johnson, J.; Lotsikas, P.; Wright, B. 2022 AAHA pain management guidelines for dogs and cats. *J. Am. Anim. Hosp. Assoc.* 2022, 58, 55–76.
- [34]. De Bellis, F.; Di Mattia, D. Approach to emergency dermatology cases in dogs and cats. *Practice* 2022, 44, 436–451.
- [35]. Gedon, N.K.Y.; Mueller, R.S. Atopic dermatitis in cats and dogs: A difficult disease for animals and owners. *Clin. Transl. Allergy* 2018, 8, 41.
- [36]. Foster, M.; Mealin, S.; Gruen, M.; Roberts, D.L.; Bozkurt, A. Preliminary evaluation of a wearable sensor system for assessment of heart rate, heart rate variability, and activity level in working dogs. In *Proceedings of the 2019 IEEE SENSORS*, Montreal, QC, Canada, 27–30 October 2019; pp. 1–4.
- [37]. Ladha, C.; Hammerla, N.; Hughes, E.; Olivier, P.; Ploetz, T. Dog's life: Wearable activity recognition for dogs. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, Zurich, Switzerland, 8–12 September 2013; pp. 415–418.
- [38]. Ahn, J.; Kwon, J.; Nam, H.; Jang, H.K.; Kim, J.I. Pet buddy: A wearable device for canine behavior recognition using a single IMU. In *Proceedings of the 2016 International Conference on Big Data and Smart Computing (BigComp)*, Hong Kong, China, 18–20 January 2016; pp. 419–422.
- [39]. Hussain, A.; Ali, S.; Kim, H.C. Activity detection for the wellbeing of dogs using wearable sensors based on deep learning. *IEEE Access* 2022, 10, 53153–53163.

- [40]. Wells, D.L. The state of research on human–animal relations: Implications for human health. *Anthrozoös* 2019, 32, 169–181.
- [41]. Seon-Chil Kim; Sanghyun Kim; Development of a Dog Health Score Using an Artificial Intelligence Disease Prediction Algorithm Based on Multifaceted Data, *Animals* 2024, 14, 256.
- [42]. Foster, M.; Brugarolas, R.; Walker, K.; Mealin, S.; Cleghern, Z.; Yuschak, S.; Clark, J.C.; Adin, D.; Russenberger, J.; Gruen, M.; et al. Preliminary evaluation of a wearable sensor system for heart rate assessment in guide dog puppies. *IEEE Sens. J.* 2020, 20, 9449–9459.
- [43]. Vikartovska, Z.; Farbakova, J.; Smolek, T.; Hanes, J.; Zilka, N.; Hornakova, L.; Humenik, F.; Maloveska, M.; Hudakova, N.; Cizkova, D. Novel diagnostic tools for identifying cognitive impairment in dogs: Behavior, biomarkers, and pathology. *Front. Vet.* 2021, 7, 551895.
- [44]. Campbell, J. Allergic skin diseases in dogs and cats. In *Clinical Atlas of Canine and Feline Dermatology*; Coyner, K.S., Ed.; Wiley & Sons: Hoboken, NJ, USA, 2020; pp. 215–253.
- [45]. Parker, V.J. Nutritional management for dogs and cats with chronic kidney disease. *Vet. Clin. Small Anim.* 2021, 51, 685–710.
- [46]. Bart, R.; Ishak, W.W.; Ganjian, S.; Jaffer, K.Y.; Abdelmesseeh, M.; Hanna, S.; Gohar, Y.; Azar, G.; Vanle, B.; Dang, J.; et al. The assessment and measurement of wellness in the clinical medical setting: A systematic review. *Innov. Clin. Neurosci.* 2018, 15, 14–23.
- [47]. Hoke, A.M.; Keller, C.M.; Grimm, C.L.; Lehman, E.B.; Sekhar, D.L. Impact of wellness policy review, wellness council activity, and student health objectives on overall school wellness climate. *J. Sch. Health* 2023.
- [48]. Navyashree S, Prof. Soniya James. Wearable Alert for Animal Health Monitoring System Using Arduino UNO. National Conference on Communication and Image Processing, 2019.
- [49]. R. Brugarolas, T. Latif, J. Dieffenderfer, K. Walker, S. Yuschak, B. L. Sherman, D. L. Roberts, A. Bozkurt, “Wearable Heart Rate Sensor System for Wireless Canine Health Monitoring”, *IEEE Sensor Journal*, May 2016.
- [50]. Brložnik, M. and Avbelj, V., 2015, May. Wireless electrocardiographic monitoring in veterinary medicine. In *Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2015 38th International Convention on (pp. 356-359). IEEE.
- [51]. Lee, H.S. A study on perception and needs of urban park users on off-leash recreation area. *KIEAE J.* 2010, 10, 49–55.
- [52]. Christian, H.; Bauman, A.; Epping, J.N.; Levine, G.N.; McCormack, G.; Rhodes, R.E.; Richards, E.; Rock, M.; Westgarth, C. Encouraging dog walking for health promotion and disease prevention. *Am. J. Lifestyle Med.* 2018, 12, 233–243.
- [53]. Boisvert, J.A.; Harrell, W.A. Dog walking: A leisurely solution to pediatric and adult obesity? *World Leis. J.* 2014, 56, 168–171.
- [54]. Enmarker, I.; Hellzén, O.; Ekker, K.; Berg, A.G.T. Depression in older cat and dog owners: The Nord-Trøndelag health study (HUNT)-3. *Aging Ment. Health* 2015, 19, 347–352.
- [55]. Lee, H.S.; Song, J.G.; Lee, J.Y. Influences of dog attachment and dog walking on reducing loneliness during the covid-19 pandemic in Korea. *Animals* 2022, 12, 483.
- [56]. Kim, S.Y.; Youn, G.H. The Relationship between pet dog ownership and perception of loneliness: Mediation effects of physical health and social support. *J. Inst. Soc. Sci.* 2014, 25, 215–233.
- [57]. Romagnoli, S.; Bensala, C.; Ferre-Dolcet, L.; Sontas, H.B.; Stelletta, C. Fertility parameters and reproductive management of Norwegian Forest Cats, Maine Coon, Persian and Bengal cats raised in Italy: A questionnaire-based study. *J. Feline Med. Surg.* 2019, 21, 1188–1197.
- [58]. Berlianti, R.; Fibriyanti, F. Perancangan Alat Pengontrolan Beban Listrik Satu Phasa Jarak Jauh Menggunakan Aplikasi Blynk Berbasis Arduino Mega. *SainETIn* 2020, 5, 17–26.

- [59]. Henny Endah Anggraeni; Aep Setiawan; Suhendi Irawan; Temperature and Humidity Monitoring System Environmental Cat Incubator Based on the Internet of Things (IoT). Proceedings 2022, 83, 50.
- [60]. Izzinnahdi, A.; Murdiantoro, R.A.; Armin, E.U. Sistem pemantauan kondisi air hidroponik berbasis internet of things menggunakan NodeMCU ESP8266. JTECE 2021, 3, 54–61.
- [61]. Rung-Ching Chen; Vani Suthamathi Saravananarajan; Hsiu-Te Hung; Monitoring the behaviours of pet cat based on YOLO model and raspberry Pi, International Journal of Applied Science and Engineering 2021
- [62]. M.F., Smith, M.L., Smith, L.N., Salter, M.G., Baxter, E.M., Farish, M., Grieve, B. 2018. Towards on-farm pig face recognition using convolutional neural networks, Computers in Industry, 98, 145–152.
- [63]. J., Divvala, S., Girshick, R., Farhadi, A. 2015. you only look once: Unified, real-time object detection, arXiv preprint arXiv:1506.02640
- [64]. Liang, C., Xiong, J., Zheng, Z., Zhong, Z., Li, Z., Chen, S., Yang, Z. 2020. A visual detection method for nighttime litchi fruits and fruiting stems, Computer and Electronics in Agriculture, 167, 105192.
- [65]. J. McNicholas and G. M. Collis, Handbook of Small Animal Dermatology The Waltham Book of Human-Animal Interaction : Benefits and Responsibilities of Pet Ownership. 1995.
- [66]. R. K. Nurlayli and D. S. Hidayati, “Kesepian Pemilik Hewan Peliharaan yang Tinggal Terpisah dari Keluarga,” J. Ilm. Psikol. Terap., vol. 02, no. 01, pp. 21–35, 2014, [Online]. Доступна на: <https://ejournal.umm.ac.id/index.php/jipt/article/view/1767/1855>.
- [67]. Karina Bella Bestari, Lis Diana Mustafa, Mochammad Junus, Design and Development of Control and Monitoring Systems in Internet of Things (IoT) Based Cat Cage (Case Study Violet Pet Shop & Clinic), Journal of Telecommunication Network (Jurnal Jaringan Telekomunikasi) Vol. 13, No. 4 (2023).

**За контакти:**

инж. Димитър В. Аврамов, докторант към  
 катедра „Компютърни науки и технологии”  
 Технически университет - Варна  
 e-mail: [dimityr.avramov.93@gmail.com](mailto:dimityr.avramov.93@gmail.com)



# IMPLEMENTATION OF A DEVELOPED MODEL FOR SENTIMENT ANALYSIS IN BULGARIAN LANGUAGE

Daniela Petrova

**Abstract:** In the author's prior research, has been developed and presented a novel ensemble method named att\_SVM+biLSTM+lex\_RF, which has demonstrated superior performance compared to other tested by the author methods in sentiment analysis of user reviews in Bulgarian. This paper presents the implementation and evaluation of this method in a web page, where everyone freely can test the method both with single comments in Bulgarian, or by uploading a file with many user reviews, to be marked and sorted as positive or negative.

**Keywords:** sentiment analysis, Bulgarian language, opinion mining, text mining, text classification, web application

## 1. Introduction

In today's digital age, where technology and the Internet are integral to daily life, the volume of web documents has surged significantly. This growth has created a critical need for methods and techniques that enable computers to understand these documents. Natural Language Processing (NLP) and its subset, Sentiment Analysis, is the key to this task. Sentiment Analysis classifies user reviews and comments into positive or negative sentiments by analyzing the opinions expressed in each sentence [1]. While there are many tools and approaches for extracting sentiment from English texts, the situation is different for less commonly used languages like Bulgarian. There are limited preprocessing tools, small datasets, and the research in this area is still in its early stages, leaving much room for development [2], [3].

Research on extracting opinions from Bulgarian texts is scarce. One of the earliest attempts at automatic classification of Bulgarian adjectives was by Boris Kraychev and Ivan Koychev [4]. They proposed a method using predefined emotional axes, such as love-hate and good-bad, and employed unsupervised training on word frequency data from Bing. Bing was chosen over Google because it provided more detailed data on word frequency and proximity [5][6]. Due to the limited availability of Bulgarian data, Kraychev and Koychev, alongside Gergana Lazarova, combined Bulgarian and English comments for analysis [7].

While not directly focused on opinion extraction, several other studies on Bulgarian texts can support sentiment analysis. For example, Tsvetana Dimitrova and Valentina Stefanova worked on the semantic classification of adjectives in Bulgarian WordNet, leveraging existing WordNet data [8]. Ivelina Stoyanova's dissertation explored automatic detection of multi-word expressions in Bulgarian, which could improve sentiment analysis performance since multi-word expressions are crucial yet hard to detect [9].

A significant advancement in analyzing Bulgarian user comments was made by Borislav Kapukaranov and Preslav Nakov [2], [10], [11]. They created a 10,000-entry database of film reviews, going beyond basic positive, neutral, and negative classifications to include 11 levels of sentiment, applying SVM for both classification and regression [2]. Nakov also developed a stemmer for Bulgarian, which was used in experimental studies and preprocessing data [10], [11].

Marinov and Nivre [12] introduced the first dependency parser for Bulgarian using the BulTreeBank corpus, although it's unclear if it was made available for broader use. Savoy [13] also created a stemmer for Bulgarian and a list of stop words, comparing his results to Nakov's [10]. However, his stemmer only handles inflectional suffixes.

The BulTreeBank team highlighted the severe lack of tools for processing Bulgarian texts, forcing them to develop many resources, including tokenizers, morpho-syntax analyzers, and dependency parsers [14]. These efforts laid the foundation for future language processing tools, which are now available through BulTreeBank’s website [15].

Another smaller Bulgarian dataset with 900 comments was developed by Tsvetanka Georgieva-Trifonova, who applied various classifiers such as NB, KNN, SVM, and deep learning methods [16].

These factors prompted the author to embark on research in sentiment analysis specifically targeting the Bulgarian language. In the initial stage, two large datasets were compiled, each consisting of around 100,000 user comments in Bulgarian. The author also developed a tailored stop words list for preprocessing. Following this, experiments were conducted using popular machine learning algorithms to identify the most effective approach for sentiment analysis in Bulgarian texts. The methods tested included Naïve Bayes, Logistic Regression, Support Vector Machines, Random Forest, and Recurrent Neural Networks.

In pursuit of a more effective algorithm, researchers introduced an improved ensemble algorithm called att\_SVM+biLSTM+lex\_RF. This approach combines two machine learning techniques, Support Vector Machines (SVM) and Recurrent Neural Networks (RNN), through a stacking ensemble method, with Random Forest (RF) as the meta-classifier. To boost predictive accuracy, an attention mechanism—which targets specific parts of the input data—is applied before SVM, and a lexicon-based method is integrated into the RF classifier [17].

As previously discussed, the author conducted experiments with widely-used machine learning algorithms to identify the most effective method for sentiment analysis in Bulgarian texts. These methods included Naïve Bayes, Logistic Regression, Support Vector Machines, Random Forest, and Recurrent Neural Networks. The experiments were carried out under various conditions: with or without stemming during preprocessing, and using either uni-grams or bi-grams. The results were promising, with accuracy rates exceeding 90%. However, the highest accuracy was achieved by the author's proposed ensemble method, which combined Support Vector Machines (SVM) and Recurrent Neural Networks (RNN) through a stacking ensemble approach, with Random Forest (RF) as the meta-classifier. To further improve predictive accuracy, an attention mechanism—focusing on specific parts of the input data—was implemented before applying SVM, and a lexicon-based approach was integrated into the RF classifier (Table 1).

**Table 1** - Results of previous calculations

<b>Classification method</b> <b>Tf-IDF</b> <b>Bi-grams</b>	<b>Evaluation</b>	<b>Dataset 1</b>	<b>Dataset2</b>
<b>Naïve Bayes</b>	<b>Without stemming</b>	0,841	0,864
	<b>Stemming</b>	0,836	0,858
<b>Logistic regression</b>	<b>Without stemming</b>	0.899	0.940
	<b>Stemming</b>	0.901	0.940
<b>Logistic regression +attention mechanism</b>	<b>Without stemming</b>	0,898	0,937
	<b>Stemming</b>	0,901	0.941
<b>Support Vector Machine</b>	<b>Without stemming</b>	0.894	0,933
	<b>Stemming</b>	0,898	0,937
<b>SVM+ attention mechanism</b>	<b>Without stemming</b>	0,899	0,939
	<b>Stemming</b>	0,901	0,940
<b>Random forest</b>	<b>Without stemming</b>	0,861	0,900
	<b>Stemming</b>	0,872	0,908



<b>LSTM</b>	<b>Without stemming</b>	0,909	0,943
	<b>Stemming</b>	0,912	0,944
<b>SVM + RF</b>	<b>Without stemming</b>	0,904	0,943
	<b>Stemming</b>	0,903	0,942
<b>SVM+LSTM</b>	<b>Without stemming</b>	0,897	0,949
	<b>Stemming</b>	0,900	0,940
<b>SVM+LSTM+ attention</b>	<b>Without stemming</b>	0,905	0,942
	<b>Stemming</b>	0,907	0,943
<b>att_SVM+biLSTM+lex_RF</b>	<b>Without stemming</b>	0.905	0.944
	<b>Stemming</b>	0.913	0,948

## 2. Realization of implementation

For the realization of the current project have been used HTML and CSS, PHP for getting the data from the web page and Python for the sentiment analysis part, using the already created ensemble method. Fig. 1 presents the overall look of the web page.

**Fig. 1.** Overall look of the web page

The web page itself is divided into two parts. First one is for sentiment analysis for one user review. Fig. 2 shows the user front end for sentiment analysis of one comment. The user is asked to input one user review in Bulgarian to be assessed. After pressing the button “Analyze” the result is shown on the screen (Fig. 3). In Fig. 3 are shown two different inputs – one positive and one negative and the result that is generated by the application.

**Fig. 2.** Sentiment analysis on one comment – front end

### Sentiment Analysis Result:

Comment: Много стило заведение с приятна атмосфера, вкусна храна и страхотно обслужване. Препоръчвам Ви го

Prediction: 1/1 [=====] - ETA: 0s 1/1 [=====] - 2s 2s/step positive

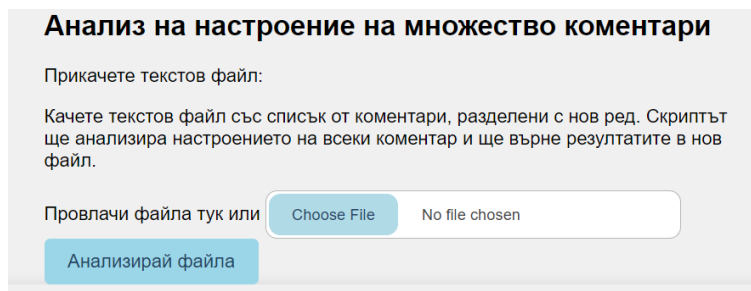
### Sentiment Analysis Result:

Comment: Клиент съм на заведението повече от две години. При последното посещение обслужването от персонала беше незадоволително внесохме торта ни маркираха два пъти такса сервиз. Барманката се държа нелюбезно.НЕ ПРЕПОРЪЧВАМ ЗАВЕДЕНИЕТО.

Prediction: 1/1 [=====] - ETA: 0s 1/1 [=====] - 2s 2s/step negative

**Fig. 3.** Results of sentiment analysis on one comment

The second part of the web page presents the option of uploading a text file with multiple user reviews to be analyzed (Fig. 4). After finishing the analysis (which is indicated by a loading spinner) the result is presented with estimation of how long each predicament took and a result file which consists of all user comments, sorted and marked as positive or negative according to the analysis. A calculation of percentage is also presented (Fig. 5).



**Анализ на настроение на множество коментари**

Прикачете текстов файл:

Качете текстов файл със списък от коментари, разделени с нов ред. Скриптът ще анализира настроението на всеки коментар и ще върне резултатите в нов файл.

Провлачи файла тук или  No file chosen

**Fig. 4.** Multiple comments sentiment analysis user input

```
1/1 [=====] - ETA: 0s
1/1 [=====] - 0s 29ms/step

1/1 [=====] - ETA: 0s
1/1 [=====] - 0s 32ms/step

1/1 [=====] - ETA: 0s
1/1 [=====] - 0s 39ms/step

1/1 [=====] - ETA: 0s
1/1 [=====] - 0s 32ms/step

1/1 [=====] - ETA: 0s
1/1 [=====] - 0s 28ms/step

1/1 [=====] - ETA: 0s
1/1 [=====] - 0s 30ms/step
Sentiment analysis completed. Results saved to c:/xampp/htdocs/python_to_php/uploads/output_file.txt.

Sentiment analysis completed. Results saved to output\_file.txt.
Positive comments: 88.79%
Negative comments: 11.21%
```

**Fig. 5.** Result of multi comments sentiment analysis

## 3. Results

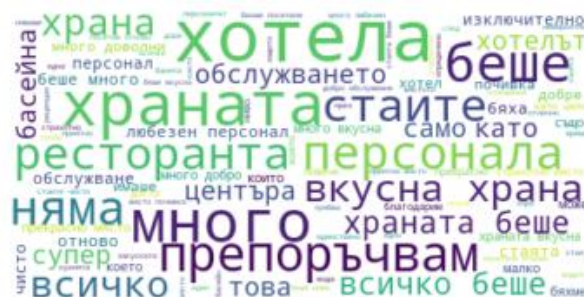
To evaluate the accuracy of the developed model and web application the author has created three datasets with user comments in Bulgarian, scraped from the web site grabo.bg. Some duplicate comments have been deleted. The initial reviews had marks from one to five, but since the model is trained to predict sentiment only positive and negative, comments with point from 1 to 3 have been marked as negative and 4 and 5 as positive.

The first dataset is for user reviews for restaurants and consists of 1 616 positive comments and 272 negative (total 1 888). Its corpus length in words is 13 212 (Fig. 6).



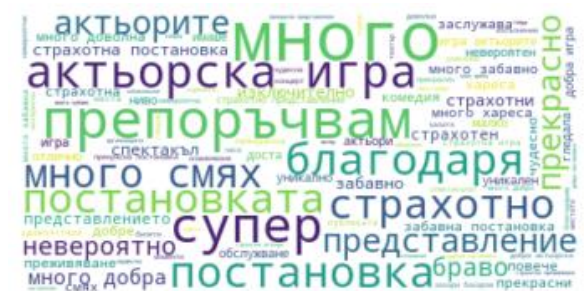
**Fig. 6.** Most common words in dataset 1

The second dataset is for comments for hotels and consists of 1 384 positive and 351 negative (total 1735) reviews. Its corpus length in words is 24 433 (Fig. 7).



**Fig. 7.** Most common words in dataset 2

And the third dataset with user comments for cinemas and theaters has 2 759 positive and 242 negative (total 3 001). Its corpus length in words is 17 536 (Fig. 8).



**Fig. 8.** Most common words in dataset 3

The results of the three datasets that have been evaluated through the suggested model are shown in Table 2. When dealing with imbalanced classes, accuracy becomes less meaningful because it equally values the model's ability to predict all categories, potentially leading to misleading conclusions and masking poor performance on the target class. In such cases, focusing solely on accuracy can obscure the model's effectiveness in identifying the specific class of interest. This is where precision and recall come into play, as these metrics specifically assess the model's performance in detecting the target class [9].

The accuracy scores indicate that the model performs well across all datasets, with Dataset 3 showing the highest accuracy at 91.63%. While accuracy is a useful overall measure, it is important to consider other metrics, especially in cases of imbalanced datasets. Precision measures the proportion of correctly identified positive comments out of all comments identified as positive. Dataset 3 exhibits the highest precision at 95.50%, indicating that the model is particularly effective in minimizing false positives in this dataset. The precision across all datasets is relatively high, showing that the model is good at correctly predicting positive sentiment.

Recall, which measures the ability to identify all actual positive comments, is consistently high across all datasets. Dataset 1 has the highest recall at 95.85%, suggesting that the model is very effective in capturing most of the positive sentiment in this dataset, though Dataset 3 is close behind. The F1 score, which balances precision and recall, reflects the model's overall effectiveness. Dataset 3 again leads with an F1 score of 95.45%, showing the best balance between precision and recall. The F1 scores across all datasets are strong, indicating that the model consistently performs well in detecting sentiment across different datasets.

Dataset 3 shows the best overall performance, with the highest scores in accuracy, precision, and F1 score, indicating that the model is particularly well-tuned for this dataset. Dataset 1 also performs well, especially in terms of recall, suggesting that it is very effective in identifying positive sentiment, though its precision is slightly lower than Dataset 3. Dataset 2, while slightly lower in all metrics compared to the other two datasets, still demonstrates solid performance, with an F1 score above 90%, indicating that the model is reliable across different datasets.

**Table 2** – Results from the implemented model

	<b>Dataset 1</b>	<b>Dataset 2</b>	<b>Dataset 3</b>
<b>Accuracy</b>	88.77%	85.76%	91.63%
<b>Precision</b>	91.44%	89.56%	95.50%
<b>Recall</b>	95.85%	92.99%	95.39%
<b>F1 Score</b>	93.60%	91.24%	95.45%

These results suggest that the model is generally robust and effective in sentiment analysis of Bulgarian user comments, with slight variations depending on the dataset. Dataset 3 appears to be the most aligned with the model's strengths, particularly in maintaining a strong balance between precision and recall.

The model's processing times for each step in this sequence range from 25ms to 63ms, with the majority of steps clustering around 44ms to 55ms. This slight variation in processing time suggests the model maintains a consistent performance, although a few outliers are slightly faster or slower. The general trend shows efficient processing, with the majority of operations completing in just over 50ms. The occasional faster step (25ms) and slower step (63ms) might be due to variations in the data or system load during processing. The consistent timing across different steps suggests stable performance without significant variations, making the model suitable for real-time or near-real-time applications where quick processing is essential.

#### 4. Conclusion

The current paper has described the creation and evaluation of web application for sentiment analysis of user reviews in Bulgarian language with a proposed from the author ensemble model called att\_SVM+biLSTM+lex\_RF. This approach combines two machine learning techniques, Support Vector Machines (SVM) and Recurrent Neural Networks (RNN), through a stacking ensemble method, with Random Forest (RF) as the meta-classifier. Additionally to enhance the predictive accuracy, an attention mechanism—which targets specific parts of the input data—is applied before SVM, and a lexicon-based method is integrated into the RF classifier. The web application works both with individual comments or multiple reviews uploaded as a text document. It has been tested with three different datasets and gives satisfactory results. The web application can be found in github and used freely.

## References

- [1]. Hajmohammadi M.S., R.Ibrahim, Z. Othman. A. Opinion mining and sentiment analysis: a survey. INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY, 2(3c), 171-178. <https://doi.org/10.24297/IJCT.V2I3C.2717>
- [2]. Kapukaranov B., P.Nakov, Fine-grained sentiment analysis for movie reviews in Bulgarian, Proceedings of Recent Advances in Natural Language Processing, p. 266-274, Hisar, Bulgaria, Sep.7-9 2015. <https://aclanthology.org/R15-1036.pdf>
- [3]. Nakov P.. (1998). BulStem: Design and Evaluation of Inflectional Stemmer for Bulgarian, Retrieved from [https://www.researchgate.net/publication/250443777\\_Design\\_and\\_Evaluation\\_of\\_Inflectional\\_Stemmer\\_for\\_Bulgarian](https://www.researchgate.net/publication/250443777_Design_and_Evaluation_of_Inflectional_Stemmer_for_Bulgarian)
- [4]. Kraychev B., Koychev I., 2012, “Computationally effective algorithm for information extraction and online review mining”, In Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics, WIMS '12, pages 64:1–64:4, Craiova, Romania
- [5]. Kraychev B., Koychev I., “Automatic classification of Bulgarian adjectives on emotional semantic axes”, VII national conference “Education and .....”p.121-130
- [6]. Крайчев Б., 2014, “Извличане и анализ на мнения и чувства от текст от онлайн източници”, Дисертация, Софийски Университет „св. Климент Охридски“, 14 януари 2014
- [7]. Lazarova G., Koychev I., (2015), “Semi-supervised Multi-view Sentiment Analysis”, In: Núñez, M., Nguyen, N., Camacho, D., Trawiński, B. (eds) Computational Collective Intelligence. Lecture Notes in Computer Science(), vol 9329. Springer, Cham. [https://doi.org/10.1007/978-3-319-24069-5\\_17](https://doi.org/10.1007/978-3-319-24069-5_17)
- [8]. Dimitrova Ts., Stefanova V., “The semantic classification of adjectives in the Bulgarian WordNet: towards a multiclass approach”, Cognitive Studies 18, Warsaw 2018 DOI: 10.11649/cs.1709
- [9]. Стоянова И., ”Автоматично разпознаване и тагиране на съставни лексикални единици в българския език“, БАН, Sofia, April 2012
- [10]. Nakov P., “2003a. Building an inflectional stemmer for Bulgarian”, In Proceedings of the 4th International Conference on Computer Systems and Technologies, CompSysTech '03, pages 419–424, Sofia, Bulgaria
- [11]. Nakov P., “2003b. BulStem: Design and evaluation of an inflectional stemmer for Bulgarian”, In Proceedings of the Workshop on Balkan Language Resources and Tools, Thessaloniki, Greece
- [12]. Marinov S., Nivre J., “A Data-Driven Dependency Parser for Bulgarian”, In: Proc. of 4th Workshop on Treebanks and Linguistic Theories (TLT'05), 2005, pp. 89-100
- [13]. Savoy J., “Searching Strategies for the Bulgarian Language”, Information Retrieval, Vol. 10, 2007, No 6, pp. 509-529. DOI: 10.1007/s10791-007-9033-9
- [14]. Simov K., Osenova P., Kolkovska S., Balabanova E., Doikoff D., “A Language Resources Infrastructure for Bulgarian”, In: Proc. of LREC'04, 2004, Lisbon, Portugal, pp. 1685-1688
- [15]. Hristova G., "Text Analytics in Bulgarian: An Overview and Future Directions" Cybernetics and Information Technologies, vol.21, no.3, 3921, pp.3-23, <https://doi.org/10.2478/cait-2021-0027>
- [16]. Georgieva-Trifonova T., Duraku M., 2021 “Research on N-grams feature selection methods for text classification”, IOP Conf. Ser.: Mater. Sci. Eng. DOI 10.1088/1757-899X/1031/1/012048
- [17]. Petrova D., V. Bozhikova (2023), „A new approach, by combination of SVM with attention mechanism and bi LSTM for sentiment analysis of Bulgarian user comments“, American

Journal of Engineering Research (AJER), e-ISSN: 2320-0847 p-ISSN : 2320-0936, Volume-12, Issue-10, pp-91-97, <https://www.ajer.org/current-issue.html>

[18]. Accuracy vs. precision vs. recall in machine learning: what's the difference? <https://www.evidentlyai.com/classification-metrics/accuracy-precision-recall>

**For contacts:**

assis. Prof. PhD Daniela Petrova  
Software and Internet Technologies  
Technical University - Varna  
E-mail: [daniela.petrova@tu-varna.bg](mailto:daniela.petrova@tu-varna.bg)



**СОФТУЕРНИ И ИНТЕРНЕТ  
ТЕХНОЛОГИИ**



# СПАМ И КИБЕРСИГУРНОСТ. СПАМ-ФИЛТРИ – ТИПОВЕ И ПРИЛОЖЕНИЕ

Стефка И. Попова

**Резюме:** С развитието на технологиите и улесняването на достъпа до интернет, все повече устройства могат да се свързват онлайн, което обаче увеличава риска от кибератаки и измами. Споделянето на лична и чувствителна информация в интернет води до нарастваща нужда от подобряване на киберсигурността. Един от най-често срещаните проблеми в областта представлява спамът, който може да бъде получен. Все по-голямата популярност на онлайн търговията и социалните мрежи води до допълнително нарастване на обема спам-съобщения. За тяхното разпознаване и отстраняване са разработени множество методи и инструменти. Тази статия прави обзор на основните видове спам и разглежда най-ефективните филтри, използвани за справяне с проблема.

**Ключови думи:** киберсигурност, спам, спам-филтри.

## Spam and cybersecurity. Spam-filters – types and usecases

Stefka I. Popova

**Abstract:** As technology advances and Internet access expands, more devices can connect online, which also increases the risk of cyberattacks and fraud. Personal and sensitive information shared on Internet leads to a growing need for improved cyber security. One of the most common problems is the distinct types of spam that a user can receive. With the steady increasing volume of spam emails, especially with the popularity of online commerce and social networks, lot of methods and tools have been developed to detect and block spam. This paper presents a review of spam types and looks at the most effective filters used to deal with the problem.

**Keywords:** cyber security, email spam, spam filters

### 1. Въведение

Улесняването на достъпа до интернет от все повече устройства, което се наблюдава през последните години, има множество предимства, но води със себе си и сериозни рискове. Споделянето на голям обем лична и чувствителна информация в мрежата доведе до увеличаване риска от кибератаки и измами, както и разпространението на зловреден софтуер. На киберсигурността се отделя все по-голямо внимание, а нейната основна задача е предотвратяването на атаки и опити за неоторизиран достъп до информационни системи, лична или чувствителна информация.

Спамът съществува в множество разновидности, от рекламни съобщения до такива, заразяващи използваната от получателя машина със зловреден софтуер. От друга страна, постоянно увеличаващата се популярност на онлайн търговията, както и нарастващият брой социални мрежи са само част от причините за все по-големия обем получаван спам по електронен път.

През годините са прилагани различни методи и са разработени множество инструменти като опит за разпознаване и отстраняване на получавания спам, но с бурното развитие на технологиите изследванията в тази област продължават.

Настоящата статия представя преглед на основните видове спам, както и на най-често прилаганите спам-филтри за разпознаване и предотвратяване получаването на спам-съобщения.

## 2. Спамът като част от киберсигурността

Киберсигурността е много широко понятие, включващо в себе си множество различни аспекти. Киберсигурността може да бъде дефинирана като защита на компютри, сървъри, мрежи, системи и данни от атаки и неправомерен достъп.

Съществуват различни класификации на типовете киберсигурност, но една от най-разпространените е тази на Европейската агенция по мрежова и информационна сигурност (European Network and Information Security Agency, ENISA), която използва модел на база заплахи. Според този модел, типовете киберзаплахи са киберактивизъм, киберпрестъпление, кибершпионаж, кибертероризъм и кибервойна [1].

Друга популярна класификация е разделянето на заплахите в пет категории: зловреден софтуер (malware), шифровачи зловредни програми (ransomware), фишинг-атаки (phishing), социален инженеринг (social engineering) и постоянна заплаха (Advanced Persistent Threat).

Спамът попада в категорията на социалния инженеринг и може да бъде използван като метод за провеждане на атаки, попадащи във всяка една от останалите категории. Това е една от причините за отделянето на специално внимание и полагането на сериозни усилия за намаляване количеството получаван спам като част от електронната кореспонденция на потребителите.

Под спам се разбира всяко получено непоискано електронно съобщение. Дадено спам-съобщение обичайно попада поне в една от следните категории: реклама, измама, съдържание за възрастни, имейл spoofing (опит за подвеждане на получателя, че са от познат източник) и други, като всяка от тях има различна цел.

Опростена структура на процеса по изпращане и получаване на електронни съобщения е посочена на фигура 1.



Фиг.1. Изпращане и получаване на електронно съобщение

Всяко едно съобщение има заглавна част, тяло, метаданни и евентуално приложения. Заглавната част съдържа информация за изпращача и получателя на съобщението, както и за евентуални допълнителни или скрити получатели, тема на писмото, неговата дата и други. Тялото на електронното писмо съдържа основната информация, която се предава и е възможно да съдържа идентификация на изпращача като име, длъжност, контакти и други. Метаданните на електронното писмо включват информация за тип съдържание, версията MIME (Multipurpose Internet Mail Extensions), използвана за форматиране на съобщението или за начина, по който текстът е кодиран.

Съществуват множество подходи и техники за минимизиране обема на получавания спам, които могат да се обособят в две групи: техники, прилагани на ниво сървър и техники, прилагани на ниво краен потребител.

Техниките, прилагани на ниво сървър, включват автентикация на изпращача чрез SPF [2], DKIM и DMARC (Sender Policy Framework, Domain Keys Identified Mail и Domain-based Message Authentication, Reporting and Conformance), DNS blacklists [3], използване на системи от тип „въпрос – отговор“ (challenge – response, като CAPTCHA) [4], контролни суми, SMTP проху, използване на различни спам-филтри и други. Тези техники и подходи обикновено имат за цел удостоверяване легитимността на изпращача.



Стандартните пощенски сървъри могат да бъдат допълнително подсигурени срещу получаване и предаване на спам чрез използване на междинни шлюзове (email gateways) или различни готови антивирусни системи и хостинг услуги като Barracuda, Cisco Email Security, Microsoft Exchange Online и множество други.

От своя страна мерките, които могат да бъдат приложени от всеки един потребител, включват използването на временни адреси при попълване на контактни форми в различни сайтове, оставяне на получено спам-съобщение без отговор, докладване или маркиране на получено електронно съобщение като спам, споделяне на адреса на електронната поща с ограничен кръг потребители и много други.

Освен прилагане на различни техники за ограничаване получаването на спам, които се прилагат на ниво сървър и от крайните потребители, доставчиците на интернет услуги също могат да използват различни методи за предотвратяване използването на тяхната услуга за изпращане на спам. Пример за подобни методи са проверка на нови клиенти, ограничаване скоростта на изпращане на съобщения, както и прилагане на изходящо филтриране.

Важно е да се посочи, че от началото на XXI-ви век много държави са приели различни законови норми като опит за борба със спам-съобщенията. В някои държави търговските съобщения задължително трябва да съдържат определен текст, което прави лесно тяхното разпознаване.

Някои от най-известните съществуващи легални мерки срещу спам са General Data Protection Regulation (GDPR), Directive on Privacy and Electronic Communications, Controlling the Assault of Non-Solicited Pornography and Marketing Act (CAN-SPAM Act) и други. Част от общите изисквания между посочените законови мерки са за налична информация за изпращача на съобщението, предварително потребителско съгласие и други.

### 3. Спам-филтри: типове и приложение

Един от най-разпространените подходи за ограничаване обема получаван спам по електронен път е използването на така наречените спам-филтри. В зависимост от типа филтър този подход може да бъде прилаган и на ниво сървър, и на ниво краен потребител.

Съществуват няколко основни групи спам-филтри, като тази част представя кратък обзор на най-разпространените от тях.

#### *Филтри на база съдържание (content-based филтри)*

Спам-филтрите на база съдържание работят на база предварително дефиниран списък с ключови думи. Всяко едно получено електронно съобщение се анализира за наличие на думи от този списък. В зависимост от честотата, с която дума от списъка се среща в конкретното писмо, се взема решение дали то да бъде класифицирано като спам или като хам (легитимно писмо). Най-често подобен тип спам-филтри се изграждат с помощта на алгоритми на машинното обучение като логистична регресия, Бейс класификатор, метод на опорните вектори, дърво на решенията, невронни мрежи и други [5, 6, 7].

#### *Филтри на база заглавна част*

Както подсказва името, подобен тип филтри анализират заглавната част на получено електронно съобщение. Анализът се фокусира върху параметри като маршрут на съобщението, IP-адрес на изпращача и неговия статус, както и дали конкретното съобщение е копие на изпратено до множество получатели писмо.

#### *Филтри на база репутация*

Спам-филтрите на база репутация работят със съществуващи списъци, съдържащи вече известни източници на спам. Те изчисляват и асоциират репутационни точки на изпращача на всяко едно електронно съобщение, базирайки се на данни от публично достъпни списъци за блокиране, различни статистики на съобщението и други параметри. При използването на този тип спам-филтър е възможно дефинирането на прагова стойност за репутационните

точки, която се явява определяща дали полученото от даден изпращач съобщение да бъде прието или отхвърлено.

#### *Филтри на база правила*

Спам-филтрите на база предварително дефинирани правила са едни от най-ефективните и често използвани филтри, заедно с тези на база анализ на съдържанието.

Както подсказва името, тези филтри оценяват всяко получено електронно съобщение на база списък от правила, като всяко правило от този списък има определен тегловен коефициент. При постъпване на ново съобщение неговия резултат е нула. След това то се проверява спрямо всяко едно правило и според тегловните коефициенти се изчислява резултата му. По подобие на репутационните филтри, при филтри на база правила също се дефинира прагова стойност, с която бива сравняван изчисления за съобщението резултат. В зависимост от това къде попада този резултат спрямо дефинирания праг, дадено съобщение се класифицира като спам или хам.

Въпреки че са много ефективни, подобен тип спам-филтри имат и недостатъци като основен недостатък се явява големият брой правила, както и техният статичен характер – те са предварително дефинирани.

#### *Филтри на база списъци*

Спам-филтрите на база списъци са друг тип широко използвани филтри. Подобен тип филтър съществува в множество разновидности и може да бъде прилаган както на ниво сървър, така и на ниво краен потребител.

Един от основно използваните типове спам-филтър на база списъци е този, базиран на „списък на отхвърляне“ (denylist filter). Той дава възможност всеки краен потребител да укаже изпращачи, които да бъдат блокирани и да не бъдат получавани съобщения от тях. Всяко едно съобщение от подобен изпращач бива автоматично отхвърлено или скрито. Ако подобен тип филтър е реализиран на сървърно ниво, източниците на съобщенията се проверяват в различни публично достъпни списъци за отхвърляне. Данните в тези публично достъпни бази от данни с източници на спам се събират с помощта на различни антиспам компании, доставчици на услуги и интернет, както и на потребители. Комбинирането на подобни списъци от двете страни допълнително повишава ефективността на филтрирането. Този тип филтри се срещат под множество други наименования – „черни списъци“ (blacklist), „списъци за блокиране“ (blocklist), „списъци за изключване“ (exclude list) и други.

Друг подобен тип спам-филтър е този, базиран на „безопасен списък“ (safelist). В него се добавят тези изпращачи, чиито съобщения да бъдат винаги пропускани през филтъра на електронни съобщения. Използването на този тип филтър предлага по-висока степен на защита в сравнение с филтър на база списък на отхвърляне. От друга страна, неговото използване прави невъзможно получаването на електронни съобщения от непознат източник, дори те да представляват легитимни съобщения. Този тип филтри са познати под имена като „бели списъци“ (whitelist), „позволен списък“ (allowlist) и други.

Съществуват спам-филтри на база списъци, които заемат междинно положение спрямо тези, използващи черни или бели списъци. Това са филтрите на база „жълти списъци“ (yellow lists), в които попадат изпращачи, чиито съобщения не са били категорично класифицирани като спам. При получаване на съобщение от изпращач, който е в подобен списък, съобщението се анализира допълнително с цел определянето му като спам или хам и предприемането на съответни мерки.

Последния тип спам-филтри, които попадат в тази категория, са тези със „сиви списъци“ (grey list). Въпреки своето име, те са по-особен случай в сравнение с посочените по-горе, защото не работят с предварително изготвени списъци. Вместо това, използването на подобен тип филтър забавя доставянето на електронно съобщение с цел удостоверяване на неговата автентичност. При постъпването на ново съобщение то се отхвърля и сървърът изпраща грешка. Подобно поведение се основава на факта, че повечето сървъри имат опция

за повторно изпращане след кратък интервал от време. Ако дадено съобщение е повторно изпратено, вероятността то да е спам не е голяма.

#### *Антивирусни филтри*

Както подсказва името на този тип спам-филтри, те разпознават, блокират и елиминират зловреден софтуер, който може да бъде изпратен чрез електронно съобщение. Всяко едно съобщение се анализира за наличие на подобен софтуер, който може да бъде скрит в неговото съдържание или да бъде прикачен като файл. Много често подобен тип филтри комбинират филтри на база съдържание и филтри на база списъци.

#### *Езикови филтри*

Последната група спам-филтри са тези на база език. Както подсказва името, използването на подобен тип филтър позволява получаването на електронни съобщения, написани на определен език. Прилагането на езиков филтър не толкова филтрира получаваните съобщения, колкото ограничава обема получавана електронна кореспонденция. Подобен тип филтри не са подходящи за потребители и организации, които имат международни контакти.

Наличието на голямо разнообразие на спам-филтри и начини за тяхното построяване дава възможност за комбинирането им в така наречения хибриден филтър. Това е и един от най-често използваните подходи за тяхното прилагане, като са проведени множество изследвания в тази посока и те продължават.

Като пример за конкретни компании, използващи хибридни филтри, могат да се посочат компаниите Google и Yahoo [8].

Google използва за предлаганата от него услуга Gmail комбинация от филтри на база съдържанието, черни и бели списъци, филтриране на изображения, сиви списъци и SPF заедно с DKIM. Анализът на съдържанието на всяко едно писмо се извършва с помощта на алгоритми на машинното обучение, като обратна връзка от потребителя повишава допълнително точността на класификация.

По подобие на Google, Yahoo също използва алгоритми на машинното обучение в комбинация с изкуствен интелект, тоест реализира филтър на база съдържанието на писмото. Отново се използват черни, бели и сиви списъци, обратна връзка, блокиране на подозрителни връзки, SPF, DKIM и DMARC.

**Таблица 1.** Типове спам-филтър и тяхната приложимост

Типове спам-филтър	Приложим на ниво пощенски сървър	Приложим на ниво краен потребител
Филтър на база съдържание	+	-
Филтър на база заглавна част	+	-
Филтър на база репутация	+	-
Филтър на база правила	+	-
Филтър на база списъци	+	+
Антивирусни филтри	+	-
Езикови филтри	+	+

Детайли по реализацията на филтрите на двете компании не са публично достъпни по очевидни причини. Посочените комбинации обаче дават достатъчна информация по отношение на използваните от тях техники и подходи. Спам-филтрите и на двете посочени компании са обучени върху големи обеми данни с използване на подходящи за подобен тип обработка софтуерни инструменти.

Таблица 1 представя обобщение на описаните по-горе типове спам-филтри и тяхната приложимост.

#### 4. Заключение

Настоящата статия представя съществуващия проблем с получаването на спам като част от електронната кореспонденция и различните подходи за неговото ограничаване. Фокусът е върху съществуващи различни видове спам-филтри, както и върху случаите на тяхното прилагане.

През годините са проведени множество изследвания с цел подобряване разпознаването на получено електронно писмо като спам. С развитието на технологиите, наблюдавано през последните десет и повече години, и с разработването на различни инструменти за работа с големи обеми от данни, изследванията в областта продължават.

#### Литература

- [1]. Lehto, M., P. Neittaanmäki. *Cyber Security: Analytics, Technology and Automation*, Springer Cham, 2015
- [2]. J. West, T. Dean, J. Andrews. *Network+ Guide to Networks (8th. ed.)*. 2018. Course Technology Press, Boston, MA, USA.
- [3]. A. Ramachandran, N. Feamster, S. Vempala. Filtering spam with behavioral blacklisting, 2007, 342-351. 10.1145/1315245.1315288.
- [4]. Tariq, N., F. Khan, S. Moqurrab, G. Srivastava. CAPTCHA Types and Breaking Techniques: Design Issues, Challenges, and Future Research Directions, 2023, 10.48550/arXiv.2307.10239.
- [5]. N. Euna, S. Hossain, M. Anwar, I. Sarker. Content-based Spam Email Detection Using an N-gram Machine Learning Approach. 2023. 10.1201/9781003256083-14.
- [6]. J. Méndez, T. Cotos-Yañez, D. Ruano-Ordás. A new semantic-based feature selection method for spam filtering, *Applied Soft Computing*, Volume 76, 2019, Pages 89-104, ISSN 1568-4946, <https://doi.org/10.1016/j.asoc.2018.12.008>.
- [7]. K. Li. Analysis of Spam Classification Based on Naive Bayes and Random Forest Model, *Advances in Economics, Management and Political Sciences*. 2024. 84. 250-257. 10.54254/2754-1169/84/20240817
- [8]. E. Dada, J. Bassi, H. Chiroma, S. Abdulhamid, A. Adetunmbi, O. Ajibuwa. Machine learning for email spam filtering: review, approaches and open research problems, *Heliyon*, Volume 5, Issue 6, 2019, e01802, ISSN 2405-8440, <https://doi.org/10.1016/j.heliyon.2019.e01802>.

**За контакти:**

Стефка И. Попова  
катедра „Софтуерни и Интернет технологии“  
Технически университет - Варна  
E-mail: [stefka.popova@tu-varna.bg](mailto:stefka.popova@tu-varna.bg)



**СОФТУЕРНИ И ИНТЕРНЕТ  
ТЕХНОЛОГИИ**

# APPLICATIONS OF MACHINE LEARNING IN TAXI BUSINESS MANAGEMENT

Todor Todorov

**Abstract:** The taxi industry, a vital component of urban transportation, faces increasing pressure from rising population densities and infrastructural demands. As cities expand, ensuring the efficiency and reliability of individual public transportation has become critical. Machine learning (ML) offers opportunities to optimize operations in the taxi sector, from dispatching to demand prediction and resource allocation. This paper explores current applications of ML technologies in taxi management and discusses their benefits, challenges, and future prospects.

**Keywords:** machine learning transport, traffic, optimization

## 1. Introduction

The taxi business is an integral part of public transportation in urban areas. In larger cities, increasing population density puts pressure on infrastructure, complicating the creation of efficient public transport systems that meet modern standards of convenience, accessibility, and efficiency. This load disrupts the balance between mass public transport (operating on schedules, such as buses, trams, metro systems, and trains) and individual public transport services like taxis, rental cars, and bike-sharing systems.

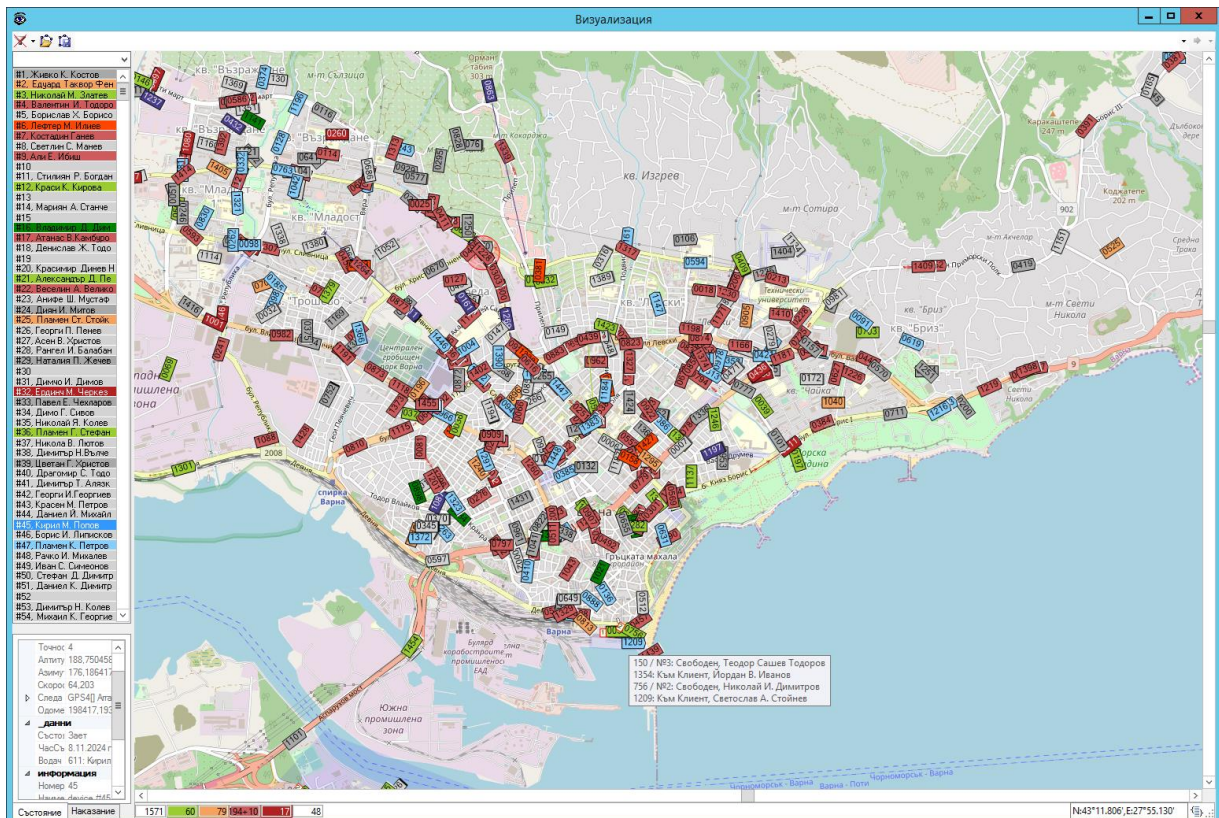


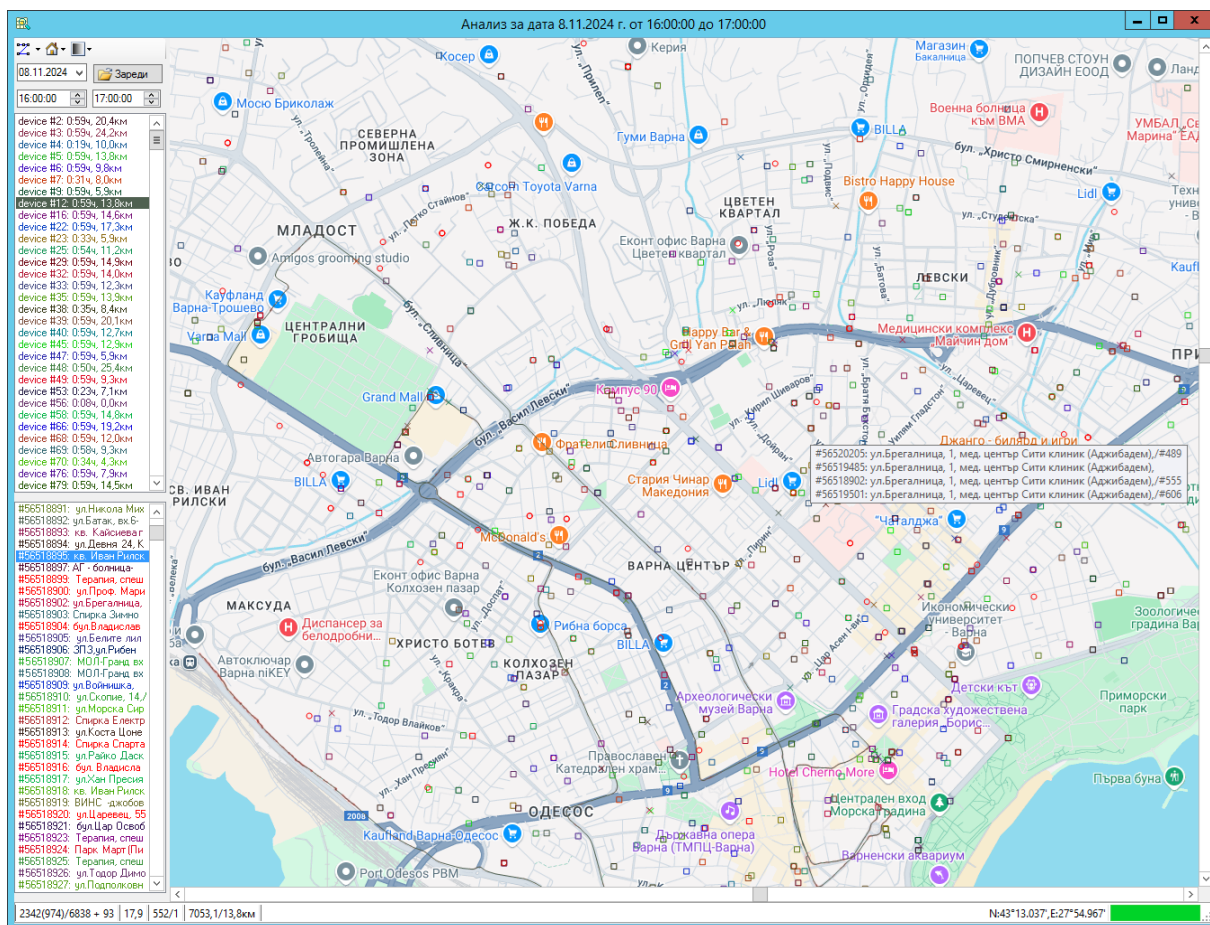
Fig. 1. Taxi company vehicle positions as of November 8, 2024, 17:53

Technological advancements, particularly in communications and GNSS positioning systems, have enabled the development of more effective software systems for transport management. By

optimizing individual travel planning, these software systems can blur the line between individual and mass transit, exemplified by the emergence of 'shared' trips that aggregate individual journeys into dynamic schedules.

Taxi services, a form of individual public transport, evolved from horse-drawn carriages and have been subject to regulations since the early 20th century. In large cities, modern taxi services operate within strict regulatory frameworks. This regulatory environment poses challenges when incorporating new technologies, as existing legislation often traces back to older forms of individual passenger transport. Despite these hurdles, many aspects of taxi operations can be significantly optimized through technology.

## 2. Specifics of Taxi System Management



**Fig. 2.** Distribution of Orders to a Taxi Company in 1 Hour

Taxi companies use real-time vehicle tracking to monitor fleet movements. The primary task of taxi management software is dispatching: registering incoming requests and assigning them to the most suitable vehicle. This process must be executed quickly, minimizing the time between receiving a customer's request and providing detailed service information, such as the vehicle and estimated arrival time.

Modern dispatch systems automate most of this process, which generally involves:

- 1) The customer defines the request: specifying pickup and drop-off locations, time, and any vehicle requirements.
- 2) The request is forwarded to available drivers.
- 3) A suitable driver is selected and asked to accept the assignment.
- 4) If the driver declines, step 3 is repeated.

5) The customer receives information about the assigned vehicle.

By automating these stages, software can handle a large volume of requests per second, making it possible to efficiently manage companies with thousands of drivers.

### 3. Case Study: Fleet Management Automation

A taxi company, founded in 2011, initially processed only 3-5% of orders through an automated system. Customers preferred interacting with a human dispatcher to feel reassured that a vehicle would be sent. By 2024, over 60% of requests are processed automatically, without human intervention, demonstrating the significant shift toward technology-driven dispatching.

### 4. Key Performance Indicators (KPIs) and Optimization

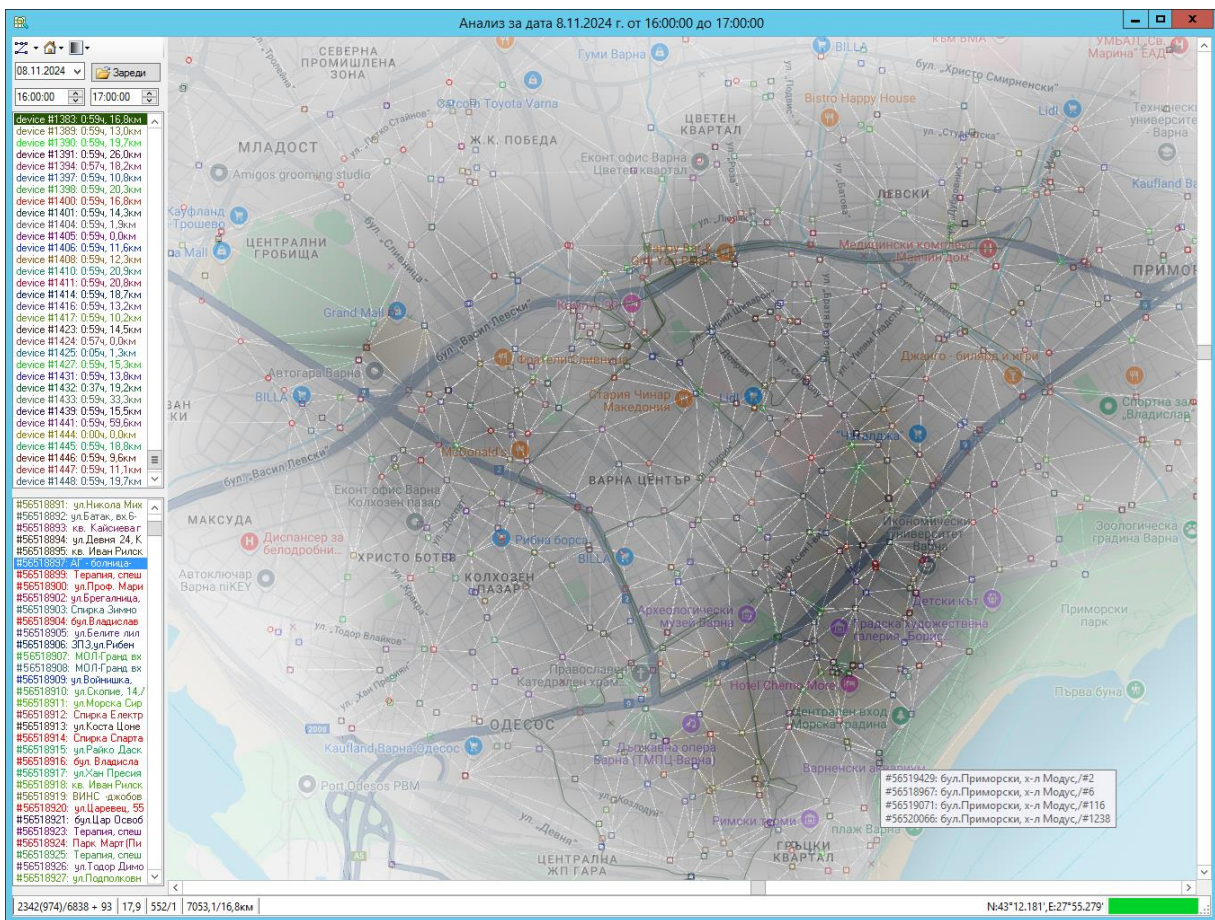


Fig. 3. Order density for the taxi company

A crucial KPI for taxi fleet efficiency is the ratio between the distance driven with passengers and the distance driven without them. Improving this ratio not only enhances financial performance but also benefits customers and the urban environment. Efficient dispatching results in higher revenue, quicker service for customers, reduced traffic congestion, and lower fuel consumption.

### 5. Predictive Modeling with Machine Learning

Predicting future demand for taxi services involves analyzing a vast amount of historical data, such as passenger pickup locations and times, and external factors like events, traffic, and

competing transport services. Using ML algorithms, companies can anticipate high-demand areas and strategically position vehicles to minimize wait times and maximize efficiency.

One example is predicting a driver's next order based on historical data. The data includes order sequence, time intervals, and the geographical coordinates of each request. By training models on this data, we can forecast the likely time and location of the next request, optimizing driver positioning.

## 6. Example Data from Taxi Orders

Data collected from drivers includes the following fields:

- no: Order sequence number
- secs\_after: Time in seconds since the previous order
- latitude, longitude: Location of the request (pickup point)

As csv-file TaxiData.json (1.jan.2023 – 0.11.2024) 25351 rows:

```
{"no":0,"secs_after":1,"latitude":43.2341842651367,"longitude":27.8762474060059 },  
{"no":1,"secs_after":38,"latitude":43.2406005859375,"longitude":27.8755035400391 },
```

...

This information forms the basis for ML models that can enhance the efficiency of taxi operations.

## 7. Comparative Models for Predicting Taxi Orders Using Historical Data

### 7.1. Methodology

We utilize various machine learning techniques to predict the three target variables defined above, using features derived from the dataset. The methodology consists of several phases:

### 7.2. Data Preprocessing

Missing values and outliers are handled to ensure data quality.

Feature engineering creates additional time-dependent features (e.g., time of day, day of the week).

### 7.3. Forecasting Techniques

Regression Models

- **Multiple Linear Regression:** Assesses relationships between multiple independent variables and the target variables.

This straightforward method attempts to establish a linear relationship between the input features and each of the output variables.

```
- import pandas as pd  
- from sklearn.model_selection import train_test_split  
- from sklearn.linear_model import LinearRegression  
- from sklearn.metrics import mean_squared_error  
-  
- # Load the dataset  
- data = pd.read_json('/content/gdrive/MyDrive/Colab  
Notebooks/files/TaxiData.json')
```



```

- # Prepare features and target variables
- X = data[['no', 'secs_after', 'latitude', 'longitude']] # Adjust
  feature names as needed
- y = data[['secs_after', 'latitude', 'longitude']]
-
- # Splitting the dataset
- X_train, X_test, y_train, y_test = train_test_split(X, y,
  test_size=0.2, random_state=42)
-
- # Initialize and train the Linear Regression model
- model = LinearRegression()
- model.fit(X_train, y_train)
-
- # Make predictions
- predictions = model.predict(X_test)
-
- # Evaluate
- mse = mean_squared_error(y_test, predictions)
- print("Mean Squared Error:", mse)
-
- # Show predictions
- print(predictions)

```

```

- Mean Squared Error: 4.044363360777796e-24
- [[ 5.          43.21789551  27.90694237]
- [ 3.          43.21893692  27.9200058 ]
- [ 1.          43.21621704  27.90197945]
- ...
- [302.         43.23361969  27.93579674]
- [298.         43.23851013  27.87818336]
- [220.         43.22582626  27.99482155]]

```

- **Random Forest Regression:** An ensemble technique capable of capturing non-linear relationships.

```

- import pandas as pd
- from sklearn.model_selection import train_test_split
- from sklearn.ensemble import RandomForestRegressor
- from sklearn.multioutput import MultiOutputRegressor
-
- # Load the dataset
- data = pd.read_json('/content/gdrive/MyDrive/Colab
  Notebooks/files/TaxiData.json')
-
- # Prepare features and target variables
- X = data[['no', 'secs_after', 'latitude', 'longitude']]
- y = data[['secs_after', 'latitude', 'longitude']]
-
- # Splitting the dataset
- X_train, X_test, y_train, y_test = train_test_split(X, y,
  test_size=0.2, random_state=42)
-
- # Initialize Random Forest Regressor for multi-output
- model = MultiOutputRegressor(RandomForestRegressor(n_estimators=100))

```

```

- model.fit(X_train, y_train)
-
- # Predictions
- predictions = model.predict(X_test)
-
- # Show predictions
- print(predictions)
-
- [[ 5.          43.21789555  27.90694254]
- [ 3.          43.21893692  27.92000583]
- [ 1.          43.21621709  27.9019795 ]
- ...
- [301.96       43.23361935  27.93579647]
- [297.95       43.23851075  27.87818172]
-
- [220.04       43.2258271   27.99482224]]

```

- **Neural Networks:** Utilizes a multi-layer perceptron (MLP) to learn complex mappings from features to targets.

```

- import pandas as pd
- from sklearn.model_selection import train_test_split
- from sklearn.preprocessing import StandardScaler
- from keras.models import Sequential
- from keras.layers import Dense
- from keras.optimizers import Adam
-
- # Load the dataset
- data = pd.read_json('/content/gdrive/MyDrive/Colab
- Notebooks/files/TaxiData.json')
-
- # Prepare features and target variables
- X = data[['no', 'secs_after', 'latitude', 'longitude']] # Adjust
- feature names as needed
- y = data[['secs_after', 'latitude', 'longitude']]
-
- # Splitting the dataset
- X_train, X_test, y_train, y_test = train_test_split(X, y,
- test_size=0.2, random_state=42)
-
- # Feature scaling
- scaler = StandardScaler()
- X_train_scaled = scaler.fit_transform(X_train)
- X_test_scaled = scaler.transform(X_test)
-
- # Initialize the MLP model
- model = Sequential()
- model.add(Dense(64, input_dim=X_train.shape[1], activation='relu'))
- model.add(Dense(32, activation='relu'))
- model.add(Dense(y_train.shape[1], activation='linear'))
-
- # Compile the model
- model.compile(optimizer=Adam(learning_rate=0.001),
- loss='mean_squared_error')

```

```

- # Train the model
- model.fit(X_train_scaled, y_train, epochs=10, batch_size=10,
- validation_split=0.1)
-
- # Make predictions
- predictions = model.predict(X_test_scaled)
-
- # Evaluate
- mse = mean_squared_error(y_test, predictions)
- print("Mean Squared Error:", mse)
- # Show predictions
- print(predictions)
Epoch 1/10
/usr/local/lib/python3.10/dist-packages/keras/src/layers/core/dense.py:87: UserWarning: Do not
pass an `input_shape`/`input_dim` argument to a layer. When using Sequential models, prefer using an
`Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
1826/1826 ----- 6s 2ms/step - loss: 193807232.0000 -
val_loss: 67572272.0000
Epoch 2/10
1826/1826 ----- 4s 2ms/step - loss: 152955632.0000 -
val_loss: 61767532.0000
Epoch 3/10
1826/1826 ----- 6s 2ms/step - loss: 149406496.0000 -
val_loss: 56891288.0000
Epoch 4/10
1826/1826 ----- 5s 2ms/step - loss: 144376352.0000 -
val_loss: 49408908.0000
Epoch 5/10
1826/1826 ----- 4s 2ms/step - loss: 139762336.0000 -
val_loss: 40230568.0000
Epoch 6/10
1826/1826 ----- 6s 2ms/step - loss: 116174152.0000 -
val_loss: 29323832.0000
Epoch 7/10
1826/1826 ----- 5s 2ms/step - loss: 68541552.0000 -
val_loss: 17631536.0000
Epoch 8/10
1826/1826 ----- 6s 3ms/step - loss: 52479756.0000 -
val_loss: 9010989.0000
Epoch 9/10
1826/1826 ----- 5s 3ms/step - loss: 24276934.0000 -
val_loss: 2560753.2500
Epoch 10/10
1826/1826 ----- 5s 2ms/step - loss: 14043693.0000 -
val_loss: 429156.2812
159/159 ----- 0s 2ms/step
Mean Squared Error: 3549207.5740146935
[[-289.66626  26.243877  21.388508 ]
 [-313.77405  22.86776  18.827885 ]
 [-253.89307  27.503845  21.384464 ]

```

```
...  
[ 97.624146 29.688152 20.580662 ]  
[ 487.42444 57.30104 32.781998 ]  
[ 519.6702 15.7127285 14.846485 ]]
```

#### 7.4. Comparative Analysis

The comparative analysis focuses on evaluating the performance and applicability of three machine learning models: Multiple Linear Regression, Random Forest Regression, and Neural Networks using a Multi-Layer Perceptron (MLP). These models are analyzed based on their prediction accuracy for forecasting taxi demand variables—order time intervals, and latitude and longitude of upcoming orders.

Multiple Linear Regression, a baseline model, leverages simplicity and interpretable coefficients to predict outcomes. It assumes linear relationships between input features and target variables, providing a clear advantage in ease of implementation and result interpretation. However, it struggles with capturing non-linear dependencies inherent in complex datasets.

The Random Forest Regression model introduces robustness and flexibility, excelling in capturing interactions between variables without requiring extensive parameter tuning. Its ensemble approach enhances predictive accuracy by averaging across numerous decision trees, thus minimizing overfitting.

Conversely, Neural Networks with an MLP architecture offer the capability to learn intricate patterns via deep layers, adaptable to non-linear and high-dimensional data. Despite their computational intensity, they demonstrate superior adaptability in revealing subtle data dynamics, providing improved prediction performance in diverse scenarios.

#### 7.5. Experimental Setup and Evaluation

The experimental setup involved splitting the dataset, "TaxiData.json", into training and testing subsets in an 80-20 ratio. Preprocessing steps ensured the standardization and normalization of continuous variables, enhancing model convergence rates and predictive stability. Each model was trained and evaluated using consistent cross-validation schemes to ensure fair performance assessment.

For the Multiple Linear Regression model, linear relationships among features were assumed, with enhancements via polynomial terms explored to capture any non-linear patterns. Random Forest Regression was employed with default parameter settings, enabling automatic feature selection and interaction modelling.

The Neural Network was configured with two dense hidden layers, utilizing ReLU activation functions to introduce non-linear computations. The model was optimized using the Adam optimizer, guided by Mean Squared Error (MSE) as the loss function, iterating through 10 epochs for convergence.

Evaluation metrics centered on comparing Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared values across models, reflecting both accuracy and explanatory power.

#### 7.6. Results and Discussion

The experimental results highlight distinctive strengths among the evaluated models. Multiple Linear Regression exhibited a strong baseline performance, delivering quick and interpretable insights but lagged in scenarios requiring non-linear extrapolation. This affirmed its utility in clear, relationship-driven predictions but underscored the need for more sophisticated approaches in complex data environments.

Random Forest Regression outperformed linear models in accuracy, attributed to its strength in managing non-linear patterns and accommodating diverse data inputs. Its ensemble design mitigated overfitting and offered balanced predictions across geographical and temporal data aspects.

The Neural Network model demonstrated the most adaptive performance, effectively capturing intricate data features with superior R-squared values, reflecting high correlation with actual observations. The MLP model's performance underlined the significance of non-linear modeling potential, particularly in spatial-temporal forecasting tasks.

In conclusion, while Multiple Linear Regression remains a valuable tool for straightforward predictive tasks, both Random Forests and Neural Networks provide substantive gains in accuracy and robustness. Neural Networks, in particular, hold substantial promise for future taxi demand forecasting, given their adaptability to varied, non-linear data structures. Planning service deployment strategies around these insights can enhance operational efficiency in taxi businesses, reducing idle times and optimizing resource allocation.

## 8. Conclusion

The employment of machine learning techniques in taxi business management can significantly elevate operational efficiency by enabling better anticipation of demand and enhanced service delivery. The ability to predict multiple outputs simultaneously assists taxi services in deploying resources more effectively throughout urban landscapes.

## References

- [1]. Barthélemy, M. Spatial networks. *Phys. Rep.* 2011, 499, 1–101.
- [2]. Cervero, R. Mixed land-uses and commuting: Evidence from the American Housing Survey. *Transp. Res.*
- [3]. Hall, C.M.; Le-Klähn, D.T.; Ram, Y. *Tourism, Public Transport and Sustainable Mobility*; Channel View Publications; Taylor & Francis: Abingdon, UK, 2017; Volume 4.
- [4]. Kivelä, M.; Arenas, A.; Barthelemy, M.; Gleeson, J.P.; Moreno, Y.; Porter, M.A. Multilayer networks. *J. Complex Netw.* 2014, 2, 203–271.
- [5]. Liu, X.; Gong, L.; Gong, Y.; Liu, Y. Revealing travel patterns and city structure with taxi trip data. *J. Transp. Geogr.* 2015, 43, 78–90.
- [6]. Markovich, J.; Lucas, K. *The Social and Distributional Impacts of Transport: A Literature Review*; Working Paper;
- [7]. *Transport Studies Unit School of Geography and the Environment, University of Oxford*: Oxford, UK, 2011. Part A: *Policy Pract.* 1996, 30, 361–377.
- [8]. Ye, X.; Huang, Q.; Li, W. Integrating big social data, computing, and modelling for spatial social science. *Cartogr. Geogr. Inf. Sci.* 2016.

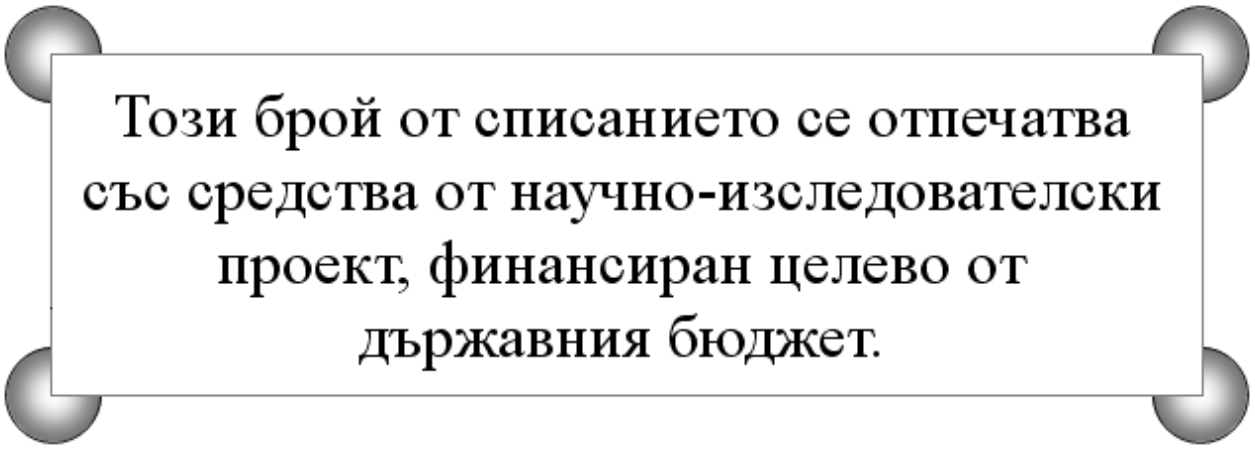
### For contacts:

Eng. Todor N. Todorov, PhD student  
Computer Science and Engineering Department  
Technical University of Varna  
E-mail: todor.n.todorov@tu-varna.bg

# ИЗИСКВАНИЯ ЗА ОФОРМЯНЕ НА СТАТИИТЕ ЗА СПИСАНИЕ "КОМПЮТЪРНИ НАУКИ И ТЕХНОЛОГИИ"

- I. Статиите се представят разпечатани в два екземпляра (оригинал и копие) в размер до 6 страници, формат А4 на адрес: Технически университет – Варна, ФИТА, ул. „Студентска” 1, 9010 Варна, както и в електронен вид на имейл адрес: [yulka.petkova@tu-varna.bg](mailto:yulka.petkova@tu-varna.bg).
- II. Текстът на статията трябва да включва: УВОД (поставяне на задачата), ИЗЛОЖЕНИЕ (изпълнение на задачата), ЗАКЛЮЧЕНИЕ (получени резултати), БЛАГОДАРНОСТИ към сътрудниците, които не са съавтори на ръкописа (ако има такива), ЛИТЕРАТУРА и информация за контакти, включваща: научно звание и степен, име, организация, поделение (катедра), e-mail адрес.
- III. Всички математически формули трябва да са написани ясно и четливо (препоръчва се използване на Microsoft Equation).
- IV. Текстът трябва да бъде въведен във файл във формат WinWord 2000/2003 с шрифт Times New Roman. Форматирането трябва да бъде както следва:
  1. Размер на листа - А4, полета: ляво - 20мм, дясно - 20мм, горно - 15мм, долно - 35мм, Header 12.5мм, Footer 12.5мм (1.25см).
  2. Заглавие на български език - размер на шрифта 16, удебелен, главни букви.
  3. Един празен ред - размер на шрифта 14, нормален.
  4. Имена на авторите - име, инициали на презиме, фамилия, без звания и научни степени - размер на шрифта 14, нормален.
  5. Два празни реда - размер на шрифта 14, нормален.
  6. Резюме и ключови думи на български език, до 8 реда - размер на шрифта 11, нормален.
  7. Заглавие на английски език - размер на шрифта 12, удебелен.
  8. Един празен ред - размер на шрифта 11, нормален.
  9. Имена на авторите на английски език - размер на шрифта 11, нормален.
  10. Един празен ред - размер на шрифта 11, нормален.
  11. Резюме и ключови думи на английски език, до 8 реда - размер на шрифта 11, нормален.
  12. Основните раздели на статията (Увод, Изложение, Заключение, Благодарности, Литература) се формират в едноколонен текст както следва:
    - a. Наименование на раздел или на подраздел - размер на шрифта 12, удебелен, центриран, един празен ред преди наименованието и един празен ред след него - размер на шрифта 12, нормален;
    - b. Текст - размер на шрифта 12, нормален, отстъп на първи ред на параграф – 10 мм; разстояние от параграф до съседните (Before и After) за целия текст – 0.
    - c. Цитиране на литературен източник - номер на източника от списъка в квадратни скоби;
    - d. Текстът на формулите се позиционира в средата на реда. Номерация на формулите - дясно подравнена, в кръгли скоби.
    - e. Фигури - центрирани, разположение спрямо текста: “Layout: In line with text”. Номер и наименование на фигурата - размер на шрифта 11, нормален, центриран. Отстояние от съседните параграфи – 6 pt.
    - f. Литература – всеки литературен източник се представя с: номер в квадратни скоби и точка, списък на авторите (първият автор започва с фамилия, останалите – с име), заглавие, издателство, град, година на издаване, страници.
    - g. За контакти: научно звание и степен, име, презиме (инициали), фамилия, организация, поделение (катедра), e-mail адрес, с шрифт 11, дясно подравнено.

Образец за форматиране можете да изтеглите от адрес <http://csejournal.cs.tu-varna.bg/>.



Този брой от списанието се отпечатва  
със средства от научно-изследователски  
проект, финансиран целево от  
държавния бюджет.