

**Міністерство освіти і науки України  
Одеський національний технологічний університет  
Вінницький національний технічний університет  
Інститут комп'ютерної інженерії, автоматизації,  
робототехніки та програмування ім.П.Н.Платонова**



## **ПРОГРАМА**

**III ВСЕУКРАЇНСЬКОЇ  
НАУКОВО – ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ  
МОЛОДИХ ВЧЕНИХ, АСПІРАНТІВ  
ТА СТУДЕНТІВ**

**«КОМП'ЮТЕРНІ ІГРИ І МУЛЬТИМЕДІА  
ЯК ІННОВАЦІЙНИЙ ПІДХІД  
ДО КОМУНІКАЦІЇ - 2023»**

**28-29 вересня 2023 р.  
ОДЕСА**

## ПРЕЗИДІЯ ТА ОРГКОМІТЕТ КОНФЕРЕНЦІЇ

### ГОЛОВА ПРЕЗИДІЇ

**Єгоров Б.В.**, Президент ОНТУ, академік НААН України, д.т.н., професор

### ЧЛЕНИ ПРЕЗИДІЇ

**Іванченкова Л.В.**, Ректор Одеського національного технологічного університету, д.е.н., професор

**Поварова Н.М.**, проректор з наукової роботи, к.т.н., доцент

### ГОЛОВА ОРГКОМІТЕТУ

**Котлик С.В.**, директор навчально-наукового інституту комп'ютерної інженерії, автоматизації, робототехніки та програмування ОНТУ, к.т.н., доц.

### ЗАСТУПНИК ГОЛОВИ ОРГКОМІТЕТУ

**Сергій Шестопапов**, к.т.н., доц., каф. Комп'ютерної інженерії, ОНТУ

### ЧЛЕНИ ОРГКОМІТЕТУ

**Олексій Извалов**, регіональний координатор Global Game Jam в Східній Європі, ETI ім.Ельворті,

**Сергій Артеменко**, зав.каф. Комп'ютерної інженерії, ОНТУ,

**Михайло Кисленко**, Unity Developer, DAL'S Games,

**Олександр Романюк**, зав.каф. Програмного забезпечення, ВНТУ,

**Ольга Чолишкіна**, директор Інституту комп'ютерно-інформаційних технологій і дизайну, МАУП,

**Олександр Терьошин**, Unity 3d developer, BlueGoji,

**Павло Івасюк**, Senior Snapchat JS Developer, BeVisioned,

**Петро Горват**, зав.каф. Комп'ютерних систем і мереж, ДВНЗ "Ужгородський національний університет".

УДК 004.01/08

Комп'ютерні ігри та мультимедіа як інноваційний підхід до комунікації - 2023 / Матеріали III Всеукраїнської науково-технічної конференції молодих вчених, аспірантів і студентів, Одеса, 28-29 жовтня 2023 р. - Одеса, Видавництво ОНТУ, 2023 р. – 270 с.

Збірник включає матеріали доповідей учасників конференції, які об'єднані за тематичними напрямками конференції.

Збірник буде корисним як для фахівців і працівників фірм, зайнятих в області розробки та просування комп'ютерних ігор, так і для викладачів, магістрів і студентів вищих навчальних закладів, які навчаються за напрямками і спеціальностями програмного забезпечення, комп'ютерних наук, комп'ютерної інженерії, прикладної математики та обробки інформації, буде корисним професіоналам у сферах гейміфікації, кіберспорту, стрімінгу, віртуальної реальності, доповненої реальності, штучного інтелекту, машинного навчання, геймдизайну, саунддизайну.

Результати досліджень у збірнику представляють собою своєрідний зріз сучасного стану справ в перерахованих галузях знань, який може допомогти як фахівцям, так і студентам університетів скласти загальну картину розвитку комп'ютерних ігор та мультимедіа та пов'язаних з ними питань.

Наукові праці згруповані за напрямками роботи конференції та наведені в алфавітному порядку прізвищ авторів.

Матеріали (тези доповідей) друкуються в авторській редакції. Відповідальність за якість та зміст публікацій несе автор.

Матеріали подано українською та англійською мовами.  
Редактор збірника Котлик С.В.

<b>Trends and prospects for the development of artificial intelligence and neural networks in the modern world.</b> Kazantsev R., Zharikov T., Kim Ye.R. (Turan University, Kazakhstan)	132
<b>Problems of evaluating and eliminating performance bottlenecks in computer games.</b> Khoshaba O.M. (Vinnitsia National Technical University)	133
<b>Research on the estimation of process modeling effort and cost.</b> Andrii Kopp, Ibrahim Dag (National Technical University «Kharkiv Polytechnic Institute»)	135
<b>Software tool for bpmn diagrams evaluation against modeling rules.</b> Andrii Kopp, Gulden Egemen (National Technical University «Kharkiv Polytechnic Institute»)	138
<b>Software tool for business process model comprehensibility assessment.</b> Andrii Kopp, Vadym Sheveliev, Yagiz Ali Turgut (National Technical University «Kharkiv Polytechnic Institute»)	141
<b>Educational school of English language.</b> Niyazdzhanov R.R., Ismailova R.T. (Turan University)	144
<b>Analysis of hard drive operating methods for gaming software.</b> Oliinik M., Khoshaba O. (National Technical University, Vinnitsia, Ukraine)	147
<b>Research application of the spam filtering and spammer detection algorithms on computer games communications.</b> Oliinyk V., Podorozhniak A., Liubchenko N. (National Technical University “Kharkiv Polytechnic Institute”)	148
<b>The impact of the development of embedded processor systems on gaming software.</b> Ovod D., Khoshaba O. (National Technical University, Vinnitsia, Ukraine)	151
<b>General methods for investigating performance bottlenecks in game software.</b> Sychenko V., Khoshaba O. (National Technical University, Vinnitsia, Ukraine)	153
<b>Increasing game software performance due to threads and processes in the Linux operating system.</b> Yavorskyi D., Khoshaba O. (National Technical University, Vinnitsia, Ukraine)	154
<b>Beam scheme development work based on arduino pro micro c using solar panel.</b> Vladyslav Yevsieiev (Kharkiv National University of Radio Electronics)	155
<b>Аналіз продуктивності мобільних застосунків на базі кросплатформених фреймворків.</b> Антонова А.Р., Очеретенко Д.В. (Одеський національний технологічний університет)	158
<b>Метод практичної побудови розпізнавача об'єктів у реальному світі.</b> Башта А.Р., Павлова О.О. (Хмельницький національний університет, м.Хмельницький)	160
<b>Розробка методу та програмного забезпечення модуля штучного інтелекту для гри "Монополія".</b> Богомазов Д.В., Кательніков Д.І. (Вінницький національний технічний університет)	162
<b>Метод організації самонавчальної комп'ютерної гри алгоритмом еволюційного навчання штучних нейронних мереж.</b> Божик І.С., Мазурець О.В., Багрій Р.О., Кліменко В.І., Тищенко О.О. (Хмельницький національний університет)	165

strengths. Modern programming languages and libraries often provide thread and process management tools to simplify the development of multithreaded and multiprocessing applications in Linux. The increase in game software performance due to threads and processes in the Linux operating system can vary significantly depending on the nature of the game, the hardware it's running on, and how well the game is optimized for multithreading and multiprocessing. Let's consider several ways threads and processes can impact game performance.

The first way is multithreading. Parallel execution based on multithreading can improve game performance by allowing specific tasks to run concurrently on multi-core CPUs. For example, game engines can use separate threads for rendering, physics simulations, and AI calculations, utilizing all available CPU cores. Responsiveness based on multithreading can help maintain a smooth and responsive user interface (UI) even when computationally intensive tasks are running in the background. This is important for preventing frame rate drops or input lag. Faster load times based on background loading of assets (textures, models, etc.) in a separate thread can reduce load times and provide a seamless gaming experience.

The second way is multiprocessing (multiple processes). Isolation, used for running game logic and other components in separate processes, can enhance stability and security. If one component crashes, it's less likely to affect the entire game. Resource management, used for multiprocessing, can help manage resource-heavy tasks that are best isolated from the primary game process, such as video encoding or complex physics simulations. This can prevent these tasks from adversely affecting game performance. Parallelism, used for specific game servers and backend processes, especially in online multiplayer games, can benefit from multiprocessing to handle concurrent connections or game instances. However, it's important to note that adding more threads or processes does not always guarantee a linear increase in performance. That is why there are challenges to consider. These problems and considerations are as follows. Synchronization overhead is that synchronization mechanisms like mutexes or semaphores may be required when multiple threads or processes need to access shared data. If not managed carefully, these can introduce overhead and potential bottlenecks. Complexity appears when multithreaded and multiprocess programming can be complex, and managing concurrency issues, such as race conditions and deadlocks, can be challenging.

For this, Amdahl's Law states that the speedup from parallelization is limited by the fraction of the program that cannot be parallelized. That is why some game tasks are inherently sequential, meaning they can't be parallelized effectively. Resource constraints are that adding more threads or processes consumes additional memory and CPU resources. Not managed well can lead to resource contention and reduced performance.

Overall, the performance gains from threads and processes in Linux for game software depend on careful optimization, hardware capabilities, and the specific demands of the game. That is why game developers often employ profiling and benchmarking tools to identify performance bottlenecks and make informed decisions about where to implement multithreading or multiprocessing to achieve the best balance between performance and resource utilization.

UDC 681.51

## **BEAM SCHEME DEVELOPMENT WORK BASED ON ARDUINO PRO MICRO C USING SOLAR PANEL**

VLADYSLAV.YEVSEIEV (vladyslav.yevsieiev@nure.ua),  
Kharkiv National University of Radio Electronics

*This work is devoted to the coverage of BEAM robotics development issues. The author provides a comparative analysis between the BEAM robot and a classic mobile robot according to various criteria and has developed a BEAM robot scheme based on Arduino Pro Micro using a Solar Panel.*

BEAM robotics (Biology, Electronics, Aesthetics, Mechanics) is an approach to creating minimalist robots that combines a biologically inspired approach with electronics, mechanics and aesthetics [1].

BEAM robots are based on the concept of using simple components and tools such as solar panels, light sensors, motors and capacitors to create autonomous robots that can interact with their environment. One of the main principles of BEAM robotics is the efficient use of energy, similar to how nature uses energy for the functioning of living organisms. Research in the field of BEAM robotics has the following promising aspects [2,3]:

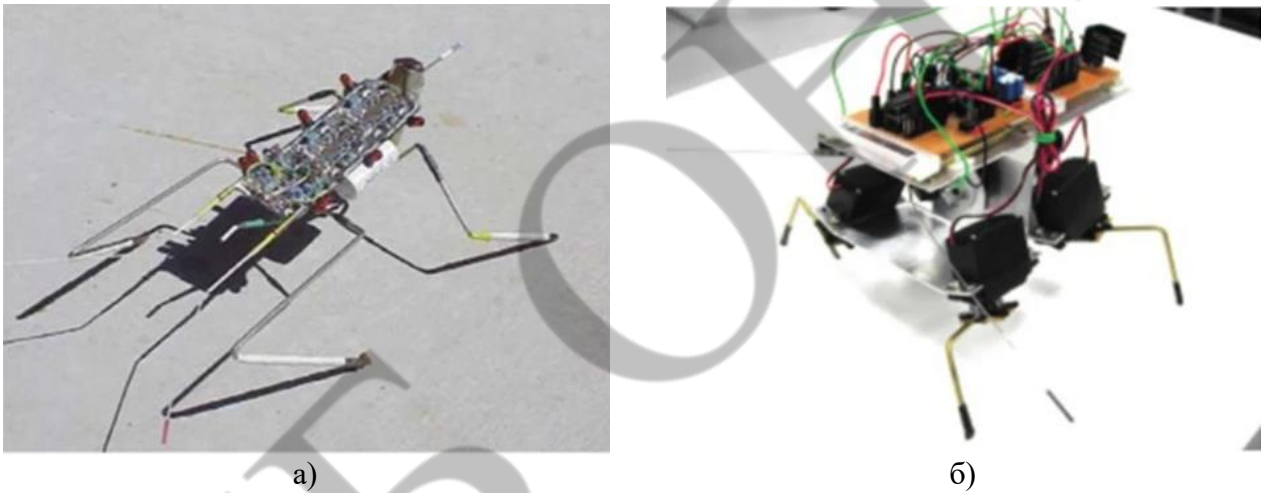
- efficient use of resources: BEAM robotics shows how functional robots can be created using minimal components and using energy from renewable sources such as solar panels. In the context of growing interest in sustainability and energy efficiency, this may have a practical application in robotics to create autonomous systems with reduced energy dependence;

- training and education: BEAM robots have long been used as a tool for training and education, as they help to experience and understand the principles of robotics, electronics and mechanics. In combination with modern distance learning platforms and virtual reality, BEAM robotics can become a powerful tool for learning and practical application of knowledge in the field of robotics;

- experimental solutions: BEAM robotics promotes a creative approach to robot construction and solving real problems. It allows you to experiment with different designs, mechanisms and electrical components to create non-standard and innovative robots;

- application in biorobotics: Insights obtained from the natural mechanisms used in BEAM robots can be useful for the development of biorobotics and biologically inspired robots. This can contribute to the creation of more adaptive and efficient robots for specific tasks [5-7].

An example of BEAM robots developed by the Mark Tilden team is presented in Figure 1.



a) insect-like shape [1]; б) QUL [1].

Figure 1 – BEAM robot developed by Mark Tilden's team

Table 1 shows a comparative analysis between the BEAM robot and the classical mobile robot according to various criteria.

Table 1 – Comparative analysis between the BEAM robot and the classical mobile robot.

Criterion	BEAM Robot	Classic Mobile Robot
Energy supply	Use of solar panels and renewable energy	Battery or accumulator
Number of components	Minimal quantity, simple components	Different components for different functions
Reaction to the environment	Reacts to changes in light, darkness, obstacles	Can use sensors to detect objects
Mechanical mechanisms	Simple mechanical moving parts	Various motors, gearboxes, wheels
Using programming	Minimal or no programming	Usually requires programming to control

Based on the selected criteria presented in Table 1, we will develop the scheme of the BEAM robot, which is presented in Figure 2.

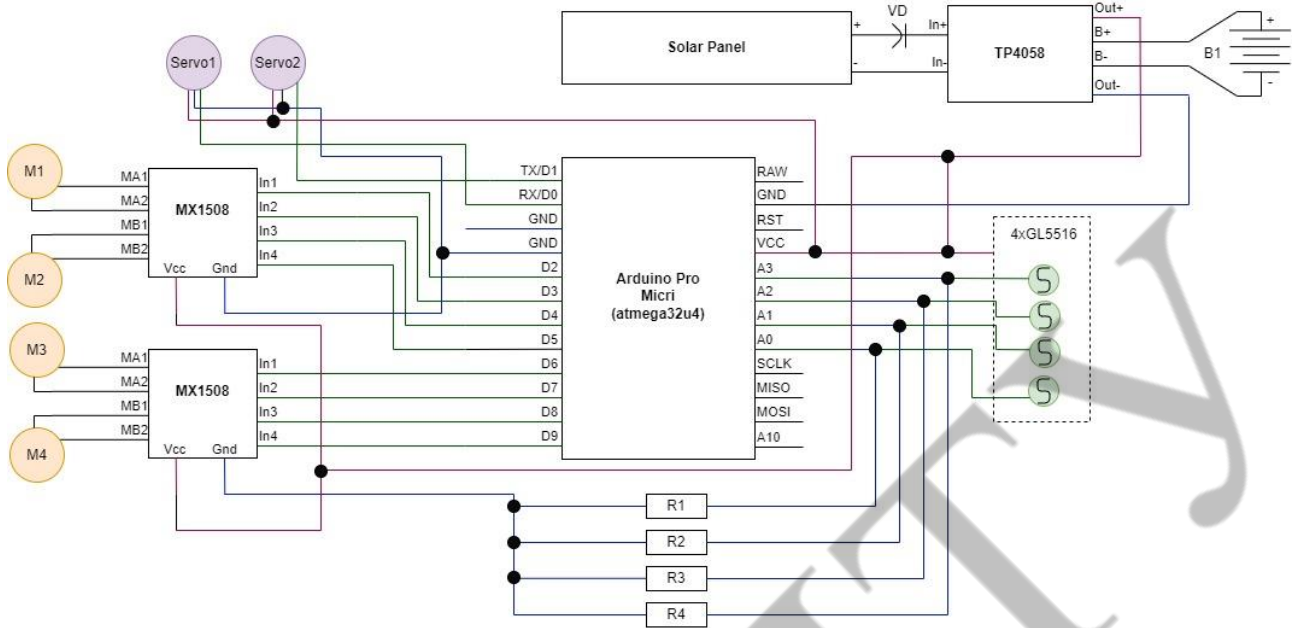


Figure 2 – Scheme of a BEAM robot based on Arduino Pro Micro using a Solar Panel

A Solar Panel (80x45mm, 5V) is used to power the BEAM robot; the TP4058 module is a charging module for lithium-ion or lithium-polymer rechargeable batteries. This module allows the battery to be charged from a power source such as a solar panel or DC source. To position the Solar Panel according to the sun, it is planned to use the GL5516 sensor and servo motors Servo1, Servo 2, which allow you to rotate the optimal angle of inclination according to the sun. The engine control system (M1, M2, M3, M4) is built on the basis of the MX1508 driver module. The general control system, information processing and decision-making system is built on the basis of the Atmega32u4 microcontroller (Arduino Pro Micro) [8-12].

**Conclusion.** All these factors show that BEAM Robotics can continue to make an important contribution to the development of robotics, even in the age of modern technology, helping to shape the approach to creating robots that interact effectively with the surrounding world and resources.

#### REFERENCES

1. Boya-Lara, C., Saavedra, D., Fehrenbach, A. et al. Development of a course based on BEAM robots to enhance STEM learning in electrical, electronic, and mechanical domains. *Int J Educ Technol High Educ* 19, 7 (2022). <https://doi.org/10.1186/s41239-021-00311-9>
2. Attar, H., & et al.. (2022). Zoomorphic Mobile Robot Development for Vertical Movement Based on the Geometrical Family Caterpillar. *Computational Intelligence and Neuroscience*, 2022, Article ID 3046116, <https://doi.org/10.1155/2022/3046116>.
3. Nevliudov, I., Yevsieiev, V., Maksymova, S., Demska, N., Kolesnyk, K., & Miliutina, O. (2022, September). Object Recognition for a Humanoid Robot Based on a Microcontroller. In *2022 IEEE XVIII International Conference on the Perspective Technologies and Methods in MEMS Design (MEMSTECH)* PP. 61-64. DOI: 10.1109/MEMSTECH55132.2022.10002906
4. Yevsieiev V. (2023) Development of a program for modeling the control of a mobile manipulation robot in the unity environment / Yevsieiev V., Starodubcev N. // *Scientific Collection «InterConf»*, (141), P. 331-334
5. Yevsieiev, V. ., Maksymova, S. ., & Starodubcev, N. . (2022). A ROBOTIC PROSTHETIC A CONTROL SYSTEM AND A STRUCTURAL DIAGRAM DEVELOPMENT. *Collection of Scientific Papers «ΛΟΓΟΣ»*, (August 12, 2022; Zurich, Switzerland), 113–114. <https://doi.org/10.36074/logos-12.08.2022.33>
6. Boya Lara, C., & Vega, M. (2020). A proposal to enhance STEM learning based on BEAM Robotics. *Publicaciones de La Facultad de Educacion y Humanidades Del Campus de Melilla*, <https://doi.org/10.30827/publicaciones.v50i4.17786>

7. Barak, M., & Assal, M. (2018). Robotics and STEM learning: students' achievements in assignments according to the P3 Task Taxonomy—practice, problem solving, and projects. *International Journal of Technology and Design Education*, 28, 121–144. <https://doi.org/10.1007/s10798-016-9385-9>
8. Cox, A. M. (2021). Exploring the impact of artificial intelligence and robots on higher education through literature-based design fictions. *International Journal of Educational Technology in Higher Education*, 18(1), 1–19. <https://doi.org/10.1186/s41239-020-00237-8>
9. Євсєєв В.В. Проектування мобільних роботів на базі одноплатних комп'ютерів (Raspberry Pi и мови Python 3.6) // Невлюдов І. Ш., Андрусевич А. О., Євсєєв В. В. Підручник. – Харків : 2020. С. 257.
10. Development and Improvement of the Design of a Lightweight Mobile Robot Manipulator Using Generative Design / I. Nevludov, V. Yevsieiev, N. Demska, H. Kostrova // Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки. Томі 34 (73) № 2. - 2023. - С.206-213.
11. Qingyi Zhang, Peng Yan, Haipeng Wang. (2022). curved-beam based quasi-constant force mechanism supporting large range and force-sensitive robotic manipulation. *Mechanism and Machine Theory*. Volume 172, June 2022, 104799. <https://doi.org/10.1016/j.mechmachtheory.2022.104799>
12. Liu, S., Li, Y. Dynamic modeling and infinite-dimensional observer-based control for manipulation of flexible beam by a multi-link robot. *Complex Intell. Syst.* 9, 3249–3260 (2023). <https://doi.org/10.1007/s40747-022-00920-5>

УДК 004.4'275

## АНАЛІЗ ПРОДУКТИВНОСТІ МОБІЛЬНИХ ЗАСТОСУНКІВ НА БАЗІ КРОСПЛАТФОРМЕННИХ ФРЕЙМВОРКІВ

АНТОНОВА А.Р., ОЧЕРЕТЕНКО Д.В.

Одеський національний технологічний університет

*Проведене дослідження присвячено кросплатформовому фреймворку Flutter. Порівняння було зосереджено на відмінностях продуктивності вбудованих програм. Дослідження показало, що Flutter-застосунки мають таку ж продуктивність, як і нативні Android-застосунки для геолокації користувача, доступу до бази даних і безперервного прокручування списків.*

З постійно зростаючими апаратними можливостями програмне забезпечення для мобільних пристроїв стає все більш складним. Незважаючи на те, що технологічні тенденції продовжують розвиватися та змінюватися з часом, проблематика розробки застосунків для певних платформ залишається незмінною, оскільки кожна платформа має власну архітектуру та набір інструментів для створення застосунків.

Існує два способи розробки мобільних застосунків: нативна розробка, націлена на конкретну операційну систему, і кросплатформова розробка, націлена на кілька операційних систем. Вибір між нативною чи кросплатформовою розробкою є одним із найважливіших рішень у будь-якому проекті мобільного застосунку. Це єдине рішення має величезні наслідки для дизайну програми, технологій, використаних для її створення, і, зрештою, користувачів, які мають до неї доступ.

Фундаментальна відмінність між нативною та кросплатформовою розробкою залежить від того, для якої операційної системи ведеться розробка. Нативна мобільна розробка дозволяє створювати програми для певної операційної системи – Android або iOS. На відміну від цього, кросплатформова розробка дозволяє створювати застосунки для кількох операційних систем.

Розробка мобільних застосунків на кросплатформових фреймворках економить час і, як наслідок, гроші на розробку. Крім того, розробникам не потрібно вивчати кожен платформу при тому, що вони все ще можуть охоплювати кінцевих користувачів цих платформ. Ще одна перевага кросплатформових фреймворків полягає в тому, що застосунок виглядатиме однаково на різних платформах. Це чудово підходить для застосунків з унікальним стилем, оскільки розробникам потрібно створити його лише один раз, як і з кодом. Однак, якщо розробник хоче, щоб застосунок виглядав як нативний, цього легше досягти за допомогою нативної системи.