

Міністерство освіти і науки України
Тернопільський національний педагогічний університет
імені Володимира Гнатюка
Інженерно-педагогічний факультет
Кафедра комп'ютерних технологій

Кваліфікаційна робота

**МЕТОДИКА ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ З ЦИФРОВИХ
ТЕХНОЛОГІЙ ДО ЗАСТОСУВАННЯ РОБОТОТЕХНІЧНИХ СИСТЕМ**

Спеціальність 015 Професійна освіта
Спеціалізація 015.39 Цифрові технології

Освітньо-наукова програма
“Професійна освіта (Комп'ютерні технології)”

ВИКОНАВ:

здобувач вищої освіти
освітнього рівня “магістр”
ГРИГОРІВ Василь Васильович

НАУКОВИЙ КЕРІВНИК:

кандидат педагогічних наук, доцент
ЯЩИК Олександр Богданович

РЕЦЕНЗЕНТ:

канд. тех. наук, доцент кафедри
машинознавства та транспорту
ТНПУ ім. В. Гнатюка

БУРЕГА Назар Васильович

Робота захищена з оцінкою:

Національна шкала _____

Кількість балів: _____ Оцінка: ECTS _____

**Тернопільський національний педагогічний університет
імені Володимира Гнатюка**
Інженерно-педагогічний факультет
Кафедра комп'ютерних технологій

ЗАВДАННЯ
ДЛЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ СТУДЕНТУ
Григорів Василю Васильовичу

на тему
**“МЕТОДИКА ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ З ЦИФРОВИХ
ТЕХНОЛОГІЙ ДО ЗАСТОСУВАННЯ РОБОТОТЕХНІЧНИХ СИСТЕМ”**

Спеціальність: 015 Професійна освіта,
спеціалізація: 015.39 Цифрові технології
Освітньо-наукова програма: Професійна освіта (Комп'ютерні
технології)

НАУКОВИЙ КЕРІВНИК: канд. пед. наук, доцент Рак Володимир Іванович

Термін подання студентом на кафедру роботи і супроводжувальних документів: до 18.05.2026 року

Зміст (перелік основних питань, які потрібно розкрити):

1. проаналізувати теоретичні засади підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем;
2. удосконалити методику вивчення робототехнічних систем в процесі підготовки майбутніх фахівців з цифрових технологій;
3. експериментально перевірити ефективність запропонованої методики в освітньому процесі.

Перелік додаткових матеріалізованих результатів роботи: розробка методичних матеріалів, конспект лекцій, лабораторні заняття, тематичне планування.

**ГРАФІК ПІДГОТОВКИ
КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

№ з/п	ПЕРЕЛІК РОБІТ	Термін виконання		Відмітка про виконання
		I рік навч. 2024-2025	II рік навч. 2025-2026	
1	Вибір теми, затвердження її на засіданні кафедри, закріплення наукового керівника	жовтень 2024		
2	Складання плану роботи і графіку її підготовки, узгодження з науковим керівником	листопад 2024		
3	Вивчення літературних і електронних джерел, збір та узагальнення фактів, даних	лютий 2025		
4	Розробка методики дослідження. Проведення пошукового дослідження	лютий 2025		
5	Написання розділу 1, подання його для перевірки керівнику	травень 2025		
6	Написання 2 розділу, подання для перевірки керівнику		грудень 2025	
7	Завершення написання роботи, оформлення її згідно з вимогами, подання науковому керівнику		березень 2026	
8	Попередній захист роботи на засіданні кафедри		квітень 2026	
9	Подання роботи на зовнішнє рецензування		травень 2026	
10	Подання кваліфікаційної роботи та супроводжувальних документів		травень 2026	
11	Захист роботи на засіданні Екзаменаційної комісії		за розкладом	

Графік узгоджено: “14” листопада 2024_ р.

Науковий керівник _____ Олександр ЯЩИК
(підпис)

Виконавець кваліфікаційної роботи _____ Василь ГРИГОРІВ
(підпис)

АНОТАЦІЯ

Григорів В. В. Методика підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем. – Магістерська робота за спеціальністю 015.39 Професійна освіта. Цифрові технології. Тернопільський національний педагогічний університет імені Володимира Гнатюка. Тернопіль, 2026. – 78 с.

У магістерській роботі обґрунтовано методику підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем.

Проаналізовано теоретичні засади підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем. Розроблено методику вивчення робототехнічних систем в процесі підготовки майбутніх фахівців з цифрових технологій. Експериментально перевірено ефективність запропонованої методики в освітньому процесі.

Робота складається з 60 сторінок основного тексту, який включає 23 рисунків, 2 таблиці, 1 діаграму та 5 додатків.

Ключові слова: роботи, робототехнічні системи, методика навчання, цифрові технології, робототехніка.

ANNOTATION

Hryhoriv V. V. Methodology for training future digital technology specialists in the application of robotic systems. – Master’s thesis in the specialty 015.39 Professional education. Digital technologies. Ternopil Volodymyr Hnatiuk national pedagogical university. Ternopil, 2026. – 78 p.

This master’s thesis sets out a methodology for training future digital technology specialists in the application of robotic systems.

It analyses the theoretical foundations for training future digital technology specialists in the application of robotic systems. A methodology has been developed for studying robotic systems as part of the training of future digital technology specialists. The effectiveness of the proposed methodology in the educational process has been experimentally verified.

The thesis comprises 60 pages of main text, including 23 figures, 2 tables, 1 diagram and 5 appendices.

Keywords: robots, robotic systems, teaching methodology, digital technologies, robotics.

ЗМІСТ

ВСТУП	8
РОЗДІЛ I. ТЕОРЕТИЧНІ ОСНОВИ ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ З ЦИФРОВИХ ТЕХНОЛОГІЙ ДО ЗАСТОСУВАННЯ РОБОТОТЕХНІЧНИХ СИСТЕМ.....	11
1.1. Огляд робототехнічних систем.....	11
1.2. Особливості вивчення робототехнічних систем при підготовці фахівців з цифрових технологій.....	19
1.3. Програмні середовища для вивчення робототехнічних систем.....	25
Висновки до першого розділу.....	35
РОЗДІЛ II. МЕТОДИКА ВИВЧЕННЯ РОБОТОТЕХНІЧНИХ СИСТЕМ В ПРОЦЕСІ ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ З ЦИФРОВИХ ТЕХНОЛОГІЙ.....	37
2.1. Вивчення робототехнічних систем на гурткових заняттях	37
2.2. Методика вивчення робототехнічних систем в цифровому середовищі Tinkercad.....	43
2.3. Апробація розробленої методики в процесі підготовки фахівців з цифрових технологій.....	50
Висновки до другого розділу	53
ЗАГАЛЬНІ ВИСНОВКИ	55
ДОДАТКИ.....	60

ВСТУП

Світ неспинно рухається до тотальної роботизації та автоматизації. Якщо раніше це спостерігалось виключно на великих підприємствах, заводах для виробництва, сортування, пакування продукції, логістики (в цьому контексті варто згадати технології Boston Dynamics), нафтовидобувній та нафтопереробній сфері та, врешті, для дослідження космосу й підвищення рівня безпеки, то зараз ми можемо спостерігати використання робототехнічних систем в побутовому вжитку. Оскільки в майбутньому здобувачі стикатимуться з такими технологіями не лише в професійній діяльності, а й в повсякденності, виникає потреба поглибленого вивчення робототехнічних систем у закладах вищої освіти.

Спостереження за тим, наскільки активно розвиваються технології та як їх використовують у все різноманітніших галузях людського життя, показує, що вже через 10 років більшість професій вимагатимуть отримання базових навичок програмування, а вміння використовувати робототехнічні технології стане вирішальним фактором при прийомі на роботу. Це означає, що вже протягом цього часу викладачам інформатики прийдеться докласти зусиль для докваліфікації чи перекваліфікації, а до навчальної програми потрібно буде внести відповідні зміни для поглибленого вивчення саме цієї сфери інформаційних технологій, адже саме вони будуть найбільше затребуваними. Бажання вдосконалювати систему освіти в Україні, а також виховувати високотехнологічне суспільство зумовило вибір відповідної теми.

Отже, робота є **актуальною**, оскільки пропонує розгляд проблем, пов'язаних з викладанням цифрових технологій, надає методичні рекомендації стосовно впровадження поглибленого вивчення робототехнічних систем в ЗВО. Правильно підібрана система допоможе викладачам відповідати професійним вимогам сучасного світу, а також краще взаємодіяти з учнем, який вже оточений технологіями, допомогти йому зрозуміти природу цих технологій, особливості їхньої роботи та вміння використовувати з метою продовження професійної діяльності.

Об’єкт дослідження: процес підготовки майбутніх фахівців з цифрових технологій.

Предмет дослідження: методика вивчення робототехнічних систем на прикладі середовища Tinkercad.

Мета дослідження: вдосконалити методику підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем та перевірити її ефективність.

Для цього було окреслено наступні **завдання дослідження:**

1. проаналізувати теоретичні засади підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем;
2. удосконалити методику вивчення робототехнічних систем в процесі підготовки майбутніх фахівців з цифрових технологій;
3. експериментально перевірити ефективність запропонованої методики в освітньому процесі.

Для досягнення мети та виконання завдань дослідження використовувалися наступні **методи дослідження:**

- аналіз теоретичної інформації та доступних наукових джерел;
- порівняння з результатами зарубіжних науковців;
- пояснення особливостей функціонування робототехнічних систем;
- створення методики з метою її подальшого дослідження чи застосування в закладах освіти.

Наукова новизна одержаних результатів полягає в обґрунтуванні методики підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем на прикладі Tinkercad.

Практичне значення роботи полягає у вдосконаленні методики вивчення робототехнічних систем в закладах вищої освіти при підготовці майбутніх фахівців з цифрових технологій.

Апробація результатів відбулась на VIII Всеукраїнської науково-практичної конференції «Актуальні проблеми та перспективи технологічної і професійної освіти» (м. Тернопіль, 23-24 квітня 2026 р.), тема доповіді:

«Підготовка майбутніх фахівців з цифрових технологій до застосування робототехнічних систем» було презентовано матеріали, що стосуються досліджуваної проблеми. [Додаток Д]

РОЗДІЛ I. ТЕОРЕТИЧНІ ОСНОВИ ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ З ЦИФРОВИХ ТЕХНОЛОГІЙ ДО ЗАСТОСУВАННЯ РОБОТОТЕХНІЧНИХ СИСТЕМ

1.1. Огляд робототехнічних систем

На сучасному етапі розвитку освіти особливу увагу приділяють інтеграції наукових та технічних дисциплін у навчальний процес. SMART-освіта визнана як ефективний метод підготовки здобувачів освіти до вимог сучасного світу. У цьому контексті освітня робототехніка відіграє важливу роль, допомагаючи створити інтерактивне та цікаве навчання з використанням роботів та технологій.

Вперше термін «робототехніка» ввів письменник Айзек Азімов ще у 1942 році. [22] Саме він сформулював 3 основні закони робототехніки, якими користуються до сьогодні:

1. Робот не може заподіяти шкоду людині або своєю бездіяльністю допустити, щоб людині була заподіяна шкода.
2. Робот повинен підкорятися командам людини, якщо ці команди не суперечать першому закону.
3. Робот повинен піклуватися про свою безпеку, поки це не суперечить першому і другому закону.

В теперішньому світі роботи тісно співпрацюють з людьми, сучасні технології дозволяють використовувати їх там, де людські можливості обмежені. Одні роботи можуть досліджувати планети, а інші використовуватися в процесі хірургічного втручання. [22]

Протягом всього існування, людина завжди прагнула полегшити свою роботу за допомогою різноманітних механізмів. Ще у стародавньому світі людство створювало примітивні машини, які могли виконувати прості функції.

Перші згадки про античні роботизовані механізми, є в епоху Стародавньої Греції та Римської імперії. [11]

Як приклад, можна навести міфічного Талоса (рис. 1.1) з острова Крит, який має детальні згадки в літописах грецької історії та давніх літературних творах. Ця легенда може бути однією з найперших згадок про створення роботів в античні часи. [23]



Рис. 1.1. Міфічний Талос з о. Крит



Рис. 1.2. Пристрій Антикера

У 1901 року між островом Крит та Кейтерою були виявлені рештки іншого дива античності – першого механізованого комп’ютера. Цей античний пристрій представляв собою складну комбінацію механізмів, що приблизно розраховували положення сонця, зірок, місяця та інших небесних тіл. Цей “комп’ютер” вважається одним із найунікальніших та найстаріших механізмів Давньої Греції і має назву “Пристрій Антикера”. (рис. 1.2) [11]

У середньовіччі, коли культура та наука переживали свій розквіт, автоматизовані годинники стали дуже популярними, особливо на дворі короля. Під час цього періоду, який припадає на час Фібоначчі та правління французького короля Людовіка IX, були розроблені одні з перших годинників з рухомими фігурами. Це були фігури ангелів, людей, тварин і навіть богів. Подекуди в цей час також згадується створення перших андроїдів у світі – механізованих фігур, які мали рухи. Один із найвідоміших андроїдів був

створений німецьким вченим Альбертом Великим. Цей андроїд схожий був на ляльку, мала зріст людини і міг відкривати та закривати двері, а також кланятися при вході. Його зовнішність і функціональні можливості нагадували поведінку людини [11]

Через два століття настала епоха активного розвитку математичних механізмів. В 1645 році французький механік Блез Паскаль винайшов відомий нам обчислювальний пристрій, який отримав назву “Паскалін”(рис. 1.3).[29]

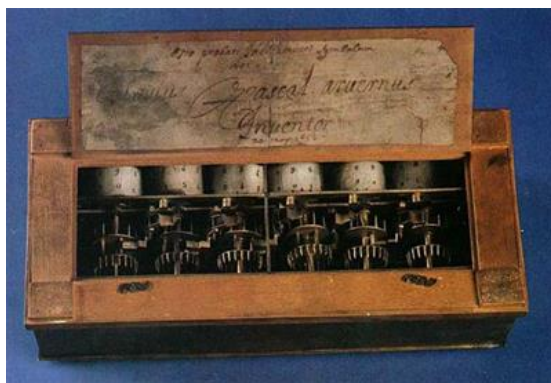


Рис. 1.3. Паскалін

У XVIII столітті мініатюрні автоматизовані пристрої стали дуже популярними як розважальні іграшки для заможних людей. Саме в цей час велику славу здобула творчість Жака де Вокансона, зокрема його найвідоміше створіння – “Качка” (рис. 1.4), яка вміла ходити, крякати та навіть дзюбати зерно. У 1801 році Джозеф-Марі Жаккард розробив ткацький верстат, який за допомогою перфокарт здатний був створювати різноманітні малюнки на тканині (рис. 1.5). У 1865 році Джон Брайнерд створив “парову людину” для транспортування колісних візків і багажу. У 1885 році Френк Ріде-молодший створив “Електричну людину” (рис. 1.6), яка була електричною модифікацією парової людини, проте була більш інноваційною та зручною в користуванні. [11]



Рис. 1.4. “Качка” Жана де Вокансона



Рис. 1.5. Перший ткацький верстат Жаккарда



Рис. 1.6. Ескіз “Парова людина”

Протягом двадцятого століття було створено безліч нових винаходів. Перший раз слово “робот” було вжито чеським письменником Карелом Чапеком у 1920 році у його п’єсі “R.U.R.” У той час цей термін позначав розумну машину, яка допомагала людині виконувати складні та нереальні завдання. У період з 1937 по 1938 рік американська компанія Westinghouse створила ELECTRO – робота, який мав здатність ходити та говорити (рис. 1.7). [11]

У 1954 році Джордж Девол розробив перший програмований робот, який обійшовся в 5 мільйонів доларів і назвав його “UNIMATE” (рис. 1.8). Цей робот використовувався для завантаження та розвантаження матеріалів, і після успішної презентації почалося масове виробництво роботів в штаті Коннектикут. [11]

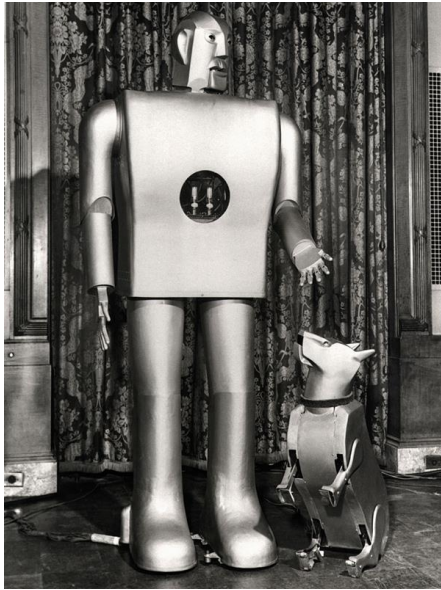


Рис. 1.7. Робот ELECTRO



Рис. 1.8. Робот UNIMATE

В 1988 року в Данбері був розроблений перший у світі медичний робот HelpMate (рис.1.9), який почали використовувати в місцевих лікарнях [11]. У 1990-х роках розпочалися роботи над створенням військових роботів. Однією з цих ініціатив стала корпорація iRobot, заснована групою механіків, включаючи Родні Брукса, Колінна Енгла і Хелен Грейнер. [11]



Рис. 1.9. Медичний робот

У 1996 році став важливим періодом для розвитку роботизованих пристроїв. В цей рік Девід Баретт розробив RoboTuna для вивчення плавання риб і інших морських істот (рис. 1.10), а компанія Honda представила робота

P2, який відзначився як перший гуманоїдний робот з двома ногами, що мав здатність до самостійного керування. [25]



Рис. 1.10. Робот RoboTuna

У 1998 році доктор Сінтія розпочав розробку концепції соціальних роботів і створив робота Kismet, який вмів взаємодіяти з людьми. Kismet (рис. 1.11) може бути розглянутий як прототип сучасного соціального робота. Зовнішній вигляд цього робота нагадує фантастичне створіння, що ніби вирвалося з казкових історій. Того ж 1998 року відома компанія “The Lego Group” випустила нову лінійку дитячих роботів MINDSTORMS, які також стали доступні на ринку. У наступному році, розробка соціальних роботів продовжила розвиватися. Компанія Sony представила свій перший робототехнічний продукт – собаку під назвою Aibo (рис. 1.12). В той же час, компанія Mitsubishi створила робота, який нагадував рибу (рис. 1.13). [11]

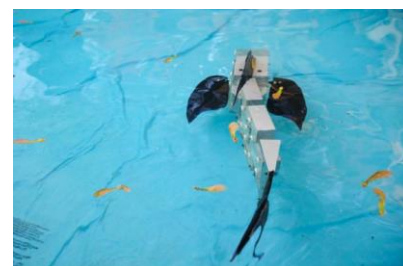
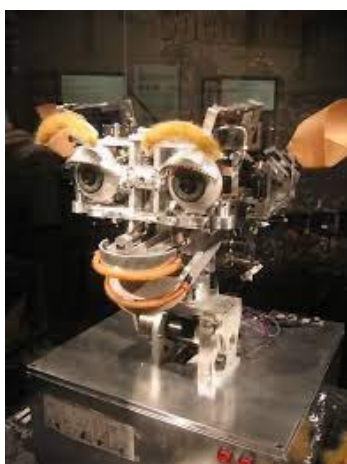


Рис. 1.11. Робот Kismet

Рис. 1.12. Робот-собака

Рис.1.13. Риба-робот

У 2000 році на виставці Robodex компанія Sony представила систему Sony Robots Dream (SDR). Ці інноваційні пристрої справді внесли значний внесок у сферу робототехніки. SDR (рис. 1.14) був обладнаний штучним інтелектом та системою аналізу, завдяки яким робот отримав низку вражаючих функцій. Він міг розпізнавати десятки різних облич, виражати власні емоції мовою тіла або мовою, а також був здатний ходити по різних типах поверхонь, будь то рівна або нерівна поверхня. [11]

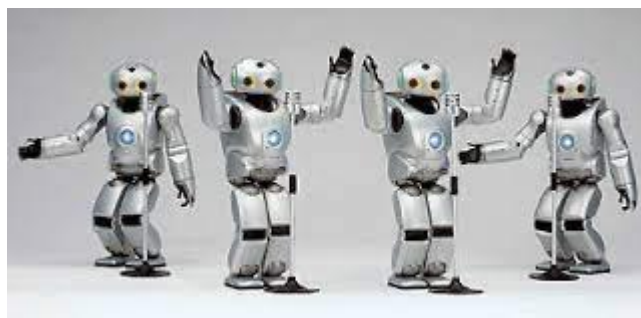


Рис. 1.14. “Роботи-мрії”

В сучасному світі інформаційних технологій та швидкого розвитку наукових досягнень, робототехніка стає все більш актуальною темою в освітньому процесі. [5]

Робототехніка, прикладна наука, що займається розробкою автоматизованих технічних систем, є одним з перспективних напрямків в галузі інформаційних технологій. Фундаментом для робототехніки слугують електроніка, механіка та програмування. Переважна більшість сучасних технічних галузей неможливо уявити без використання роботизованих систем виробництва. В свою чергу, розвиток таких галузей виробництва потребує підготовки кваліфікованих фахівців. Це, безумовно, ставить нові завдання перед сучасною системою освіти. Саме тому робототехніка стала одним із самих популярних напрямків освіти. [5]

Освітній потенціал робототехніки, як напрям SMART-освіти, надзвичайно високий. Здобувачі окрім цифрових технологій, вивчають основи робототехніки, програмування, створюючи та програмуючи власних роботів. На заняттях використовують, за наявності, специфічне технологічне

лабораторне та навчальне обладнання – 3D-принтери, засоби візуалізації та інше.

Методика та технології використання робототехніки в навчанні не є предметом цілеспрямованих вітчизняних педагогічних досліджень. Певні елементи робототехніки можливо включати як окремі розділи інформатики. Вивчення робототехніки в межах даної дисципліни може ґрунтуватися на використанні спеціалізованих конструкторів, що мають програмуємий модуль. Переважна більшість публікацій з даного питання стосується використання конструкторів лінійки LEGO Education в навчальному процесі. Робототехнічний конструктор Lego Mindstorms, Wedo, навчальні набори робототехніки UNO Robot, конструктор «Машини і механізми» для професійної освіти допомагають організувати захоплюючий практико-орієнтований освітній процес, який знайомить студентів з SMART компетенціями. [1]

Враховуючи все вище зазначене, можна зробити висновок про те, що на ринку наявна низка конструкторів, які розраховані на здобувачів освіти та сприяють розвитку навичок з робототехніки. Лише вдало розроблена методика та використання відповідних технічних засобів в процесі навчання робототехніки в ЗВО, у відповідності до мети та цілей навчання, дозволить формувати у студентів відповідні навички програмування, стимулює зацікавленість до техніки та моделювання, сприятиме розвитку логічного та алгоритмічного мислення. [20]

Робототехніка використовується в сучасних навчальних закладах для залучення здобувачів до навчання.[1] Її вплив на процес освіти полягає в наступних аспектах:

- Розвиток креативності та проблемного мислення: робототехніка допомагає студентам розвивати навички розв'язання складних завдань та стимулює їх креативність.

- Залучення до SMART-освіти: робототехніка сприяє підвищенню інтересу здобувачів до науки, технології, інженерії та програмування.
- Підготовка до майбутньої роботи: навчання робототехніці допомагає студентам отримати навички, які можуть бути корисними в майбутній кар'єрі.

Робототехніка суттєво змінює традиційні методи навчання.

Впровадження робототехніки в освітній процес призводить до таких змін:

1. Інтерактивне навчання: студенти можуть взаємодіяти з роботами, що створює активне навчальне середовище.
2. Розвиток командних навичок: робототехніка сприяє розвитку командної роботи та співпраці між студентами.
3. Можливості для індивідуалізації навчання: роботи можуть бути налаштовані для відповідності потребам кожного студента.

Робототехніка дійсно є сучасним освітнім трендом, який вносить великий внесок у покращення процесу навчання та підготовку здобувачів освіти до викликів майбутнього. Вона сприяє розвитку різних навичок та забезпечує студентам можливість вивчати науку та технологію у захоплюючий та цікавий спосіб. Робототехніка має потенціал змінити обличчя сучасної освіти і підготувати нове покоління до складних завдань і викликів, з якими вони зустрінуться в майбутньому. Зростаючий інтерес до робототехніки у навчальних закладах свідчить про її важливість у сучасному освітньому процесі. [12]

1.2. Особливості вивчення робототехнічних систем при підготовці фахівців з цифрових технологій

Робототехнічні системи – це комплекси обладнання, програмного забезпечення і механічних/електронних компонентів, призначені для

виконання автоматизованих завдань та операцій. Вони включають в себе: роботів, дронів, промислових маніпуляторів, автономних автомобілів та інші пристрої, які можуть виконувати фізичні або інтелектуальні завдання. [12]

Робототехнічні системи можуть бути використані в різних галузях, включаючи промисловість, медицину, освіту, науку, військову промисловість тощо, а також у повсякденному житті. Вони дозволяють підвищити продуктивність, покращити якість роботи та знизити ризик при виконанні різноманітних завдань. Робототехнічні системи можуть бути автономними, тобто працювати без прямого втручання людини, або контролюватися оператором. [20]

На основі проведеного огляду історії розвитку робототехніки була запропонована класифікація роботів, яку показано на рисунку 1.15. На сучасному етапі розвитку робототехніки існує шість основних груп роботів: соціальні роботи, промислові роботи, бойові роботи, побутові роботи, дослідницькі роботи, медичні роботи. Кожна з цих головних груп може мати свої власні підгрупи та споріднені складові, що відображають різноманітність застосувань робототехніки в різних галузях. [20]

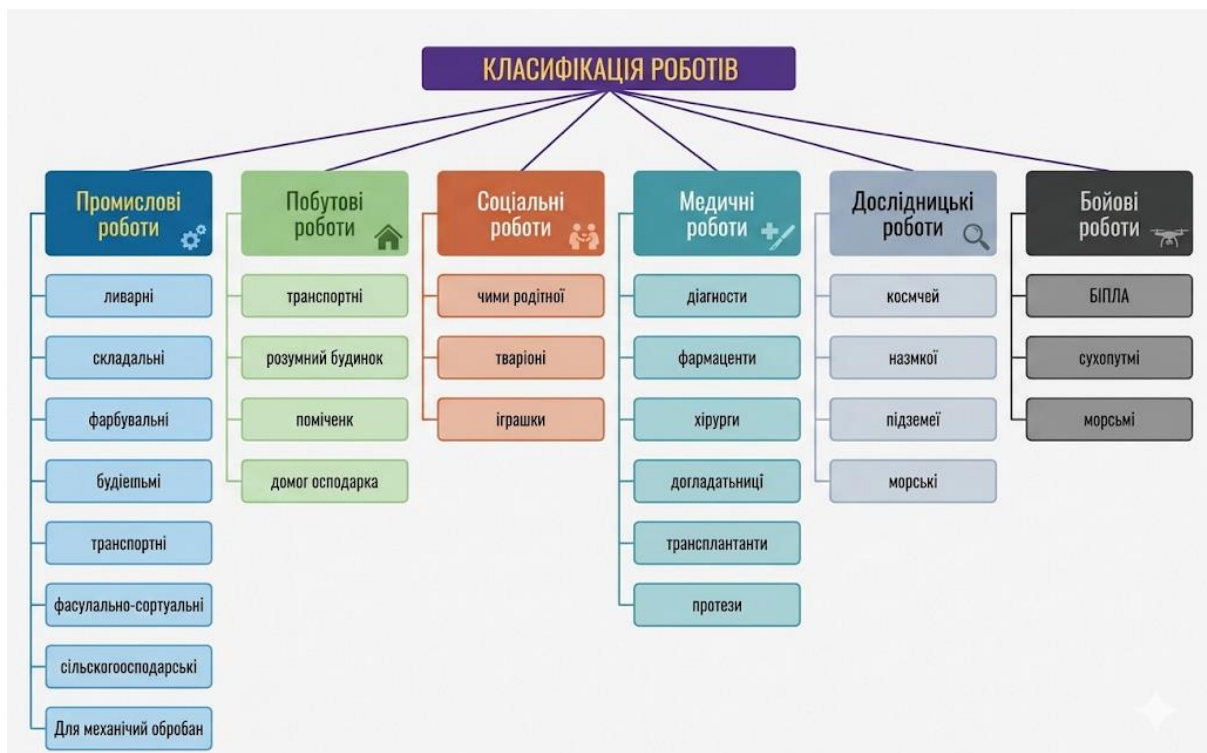


Рис. 1.15. Класифікація роботів

Промислові роботи є найпоширенішими роботами в світі, особливо в виробничій сфері. Вони спеціально розроблені для виконання різних рухомих і керуючих функцій в виробничих процесах. Ці роботи є автономними пристроями, які складаються з механічних маніпуляторів та програмованих систем управління. Основною метою промислових роботів є виконання завдань, пов'язаних з транспортуванням та обробкою різних об'єктів у виробничому просторі. Вони дозволяють автоматизувати та оптимізувати виробничі процеси, підвищуючи продуктивність та забезпечуючи високу якість виробництва. Промислові роботи використовуються в різних галузях промисловості, включаючи автомобільну, електронну, металургійну, харчову та інші сфери. [20]

Побутові роботи призначені для використання в побуті та надання допомоги людям у повсякденному житті. Вони можуть виконувати різні корисні функції, такі як прибирання, приготування їжі, пирососіння, миття посуду та інші повсякденні завдання.

Соціальні роботи здатні взаємодіяти з людьми в автономному або напіваавтономному режимах, надаючи допомогу у отриманні інформації у державних установах, громадських організаціях або на публічних місцях. Вони можуть виконувати функції, пов'язані з обслуговуванням та наданням інформації. [20]

Медичні роботи спеціалізуються на виконанні медичних та фармацевтичних процедур або операцій, пов'язаних із здоров'ям людей та медичною допомогою. До цієї категорії можуть належати роботи-хірурги, роботи-фармацевти, роботи-аптечки та інші медичні пристрої. [20]

Дослідницькі роботи призначені для проведення різноманітних досліджень та дослідницьких завдань. Вони можуть використовуватися для наукових досліджень, дослідження небезпечних або важкодоступних місць, а також для збору даних та виконання експериментів. [20]

Бойові роботи мають багатофункціональні технічні характеристики та можуть використовуватися для військових цілей. Вони можуть виконувати

різні завдання, такі як зменшення втрат людей в бойових діях, розмінування, проведення розвідки або активну участь у військових операціях. Бойові роботи можуть бути важливими для забезпечення безпеки та ефективності військових дій. [20]

Освітні роботи. Сучасні робототехнічні системи використовуються в освіті для навчання SMART-дисциплін (наука, технологія, інженерія та програмування). Вони допомагають студентам навчатися через практичний досвід та розвивають їхні навички в області програмування, робототехніки та творчого мислення. [20]

Навчання робототехніки спрямоване, перш за все, на розвиток:

- креативності;
- комунікативних навичок;
- навичок роботи в команді та над проєктами.

Основним навчальним принципом робототехніки є принцип «Навчання через дію». Навчання через дію відбувається тоді, коли студент створює реальні речі в матеріальному світі і одночасно набуває знань. [18]

Тобто, знання створюються в результаті діяльності. Відбувається циклічний процес: заново набуті знання дозволяють студенту створювати ще більш складні речі в реальному світі, які в свою чергу приносять додаткові знання, і так далі по циклу.

Кожне завдання реалізує циклічну модель (рис.1.16), яка базується на чотирьох освітніх складових: взаємозв'язку, конструюванні, рефлексії та розвитку. [2]



Рис. 1.16. Модель освітніх складових

Взаємозв'язок – ключовий принцип навчання через дію. Досліджено, що студенти краще навчаються, якщо вони можуть поєднати новий досвід з уже набутих або зробити його стимулом до нового етапу навчання та пізнання. На цьому етапі здобувачі отримують завдання з реального життя. [2]

Конструювання – цей принцип передбачає і створення моделей, і генерування ідей. Студенти поєднують знання та розуміння. Їм пропонуються особливі завдання, які заохочують їх планувати та після цього створювати моделі власної конструкції, які здатні вирішити поставлену задачу. [2]

Рефлексія – осмислення того, що зроблено, створено, модифіковано; пошук словесного формулювання отриманого знання, способів представлення результатів набутого досвіду, шляхів його застосування в комплексі з іншими ідеями та рішеннями. [2]

Розвиток – підтримка творчої атмосфери, задоволення від успішно виконаної роботи реалізуються при виконанні більш складних завдань. Це сприяє поглибленню отриманого досвіду, розвитку творчих та дослідницьких навичок. [2]

Робототехніка – це прикладна наука, що стрімко розвивається сьогодні та привертає увагу людей в усьому світі. Студентам цікаво отримувати нові знання та конструювати нові механізми, це дає потенціал, який можна використовувати у промислових масштабах.

Сучасна освітня система України стоїть перед важливим завданням адаптуватися до потреб сучасного суспільства та глобальних технологічних змін. Впровадження робототехніки як окремої дисципліни в ЗВО відіграє важливу роль у цьому процесі.

На початку 2020-х років в Україні спостерігалася активна тенденція до розвитку SMART-освіти, що включає в себе науку, технології, інженерію та програмування. Впровадження цих принципів в навчальні програми спрямоване на підготовку студентів до сучасного технологічного світу, в якому робототехніка відіграє важливу роль. Державні ініціативи сприяють розвитку SMART-освіти в Україні, а це відображається і в впровадженні робототехніки. [1]

Робототехніка в ЗВО може бути впроваджена як окрема дисципліна або як частина інших предметів. Це дозволяє студентам ознайомитися з різними аспектами робототехніки, від конструкції роботів до програмування їх рухів. Деякі ЗВО вже включають робототехніку до навчальних планів і програм, забезпечуючи студентам можливість активно вивчати цей предмет. [7]

Незважаючи на позитивні зрушення впровадження робототехніки в ЗВО, існують виклики, такі як нестача кваліфікованих викладачів та фінансування для обладнання. Однак розвиток цієї галузі продовжується, і вона має значущий потенціал для покращення якості освіти та підготовки молодого покоління до викликів майбутнього. [20]

Узагальнюючи, впровадження робототехніки як дисципліни в закладах вищої освіти є важливим компонентом розвитку SMART-освіти та підготовки студентів до сучасного технологічного світу. Державні програми та ініціативи сприяють цьому процесу, хоча існують виклики, які потребують подальшого вирішення. Перспективи впровадження робототехніки в вищу освіту обіцяють

позитивні зміни у підготовці молодого покоління до сучасного світу технології та інновацій.

У багатьох країнах існують програми та ініціативи з викладання робототехніки в ЗВО. Це допомагає студентам розвивати навички критичного мислення, працювати в команді, та готує їх до майбутніх професій у галузі науки та технологій. Заняття робототехніки також можуть бути цікавими та захоплюючими для студентів, що сприяє підвищенню зацікавленості в навчанні.

1.3. Програмні середовища для вивчення робототехнічних систем

Проаналізований іноземний досвід вказує на те, що в багатьох європейських країнах впровадження робототехніки в навчальні програми не є систематичним. Проте важливо відзначити, що востаннє велику роль у популяризації освітньої робототехніки серед студентів відіграли багаточисленні робототехнічні конструктори, які були розроблені у 2000-х роках і вдосконалені на сьогоднішній день, такі як LEGO Mindstorms NXT, Arduino, Crickets та інші. Ці конструктори створили підґрунтя для популяризації робототехніки серед студентів і студентів та сприяють розвитку цієї галузі навчання. [19]

Іспанія: Компанія Crea Robotics Education (CREA) організовує позааудиторні курси, літні заходи з освітньої робототехніки та 3D-друку. Вони також проводять підготовку інструкторів. Компанія співпрацює з Roboamreones – студентським робототехнічним турніром. [19]

У **Чехії** проводилося дослідження щодо впровадження освітньої робототехніки в навчання майбутніх викладачів інформаційно-комунікаційних технологій (ІКТ). Було розроблено навчальний план курсу освітньої робототехніки на основі конструктивістського підходу в освіті. Заняття з освітньої робототехніки проводилися в спеціалізованих ІКТ-

лабораторіях Результати дослідження підтвердили, що навчальні проекти з робототехніки сприяють розвитку технологічних знань та навичок програмування серед студентів закладів освіти. Використання робототехніки у навчанні покращує якість навчального процесу в експериментальних закладах освіти. [19]

У **Швейцарії** було введено декілька проектів з впровадження робототехніки в освіту. Один із таких проектів, з назвою “Роботи в групі”, спрямований на навчання викладачів та мотивацію їх впроваджувати робототехніку в закладах освіти. Інший проект, відомий як PReSO, був пілотним проектом з впровадження освітньої робототехніки в заклади освіти Південної Швейцарії. Його метою було підвищити інтерес здобувачів освіти до вивчення інформаційно-комунікаційних технологій (ІКТ) та SMART-предметів за допомогою робототехніки. Проте, впровадження робототехніки в заклади освіти виявилось складним завданням через відсутність відповідних навчальних матеріалів та страх викладачів перед використанням робототехніки. Для подолання цих труднощів, дослідницька група розробила концепцію підготовки викладачів до викладання робототехніки та навчила 17 викладачів. Позитивні результати проекту видно в тому, що більшість викладачів включили робототехніку до своїх навчальних програм.[19]

Австрія та Німеччина не мають систематичного впровадження робототехніки в навчальні програми, і навчання в цих країнах, в основному, організоване у формі неформальної освіти для студентів та викладачів. В деяких закладах освіти починають інтегрувати робототехніку у звичайні навчальні програми. [19]

Схожий підхід до використання робототехніки в освіті існує в **Словаччині**. Наприклад, деякі освітні заклади вирішили включити модуль робототехніки в навчальну програму з інформатики після організації позааудиторного клубу з робототехніки. У цьому модулі студенти вивчають основи робототехніки, побудову моделей з датчиками, програмування роботів

для футбольних ігор тощо. Однак заклади освіти в Словаччині ще не готові до широкого впровадження робототехніки в навчальний процес.[19]

США: Сполучені Штати запровадили різні програми та ініціативи для SMART-освіти, включаючи робототехніку. Популярні платформи, такі як VEX Robotics і For Inspiration and Recognition of Science and Technology (FIRST), пропонують змагання та навчальні програми для студентів. [19]

Японія: Японія відома своєю довгою історією в розробці роботів та робототехніки. В японських закладах освіти здобувачі можуть навчатися програмуванню та робототехніці вже з раннього віку. Клуби з робототехніки допомагають розвивати інженерні навички. [19]

Фінляндія: Фінляндія визначається високою якістю освіти. Тут робототехніка включена в навчальні програми з підходу “навчати граючись”. студенти можуть створювати та програмувати роботів в різних предметних областях, включаючи програмування та фізику. [19]

Сінгапур: У Сінгапурі навчання робототехніці в закладах вищої освіти включено до навчальних планів. студенти мають можливість розвивати навички програмування та конструювання роботів через різні платформи та робототехнічні клуби. [19]

Швеція: Швеція активно просуває SMART-освіту та робототехніку в університетах, надаючи викладачам доступ до навчальних ресурсів та підтримку для інтеграції робототехніки в навчальний процес. [19]

У **Китаї** робототехніка має величезну популярність, і на матеріально-технічне оснащення виділяється значна сума грошей. В 2014 році в місті Сіань була створена компанія China RobotC на основі програми, розробленої Інститутом Карнегі-Меллона у США. Одним із головних завдань компанії є навчання учнів та студентів робототехніки, а також підготовка викладачів з робототехніки для шкіл, коледжів, закладів професійної та вищої освіти. З метою навчання робототехніки у школах і коледжах уряд Китаю від 2018 року впровадив підручник з основами штучного інтелекту. Пілотний проект

запроваджується у 40 середніх школах і передбачає вивчення цієї дисципліни в курсі початкової та середньої освіти. [19]

Платформи бувають закритими, наприклад такі як Lego, Fischertechnik і відкритими – Arduino, Multiplo. Також можна відокремити проміжні варіанти – наприклад Huna. Особливостями закритих платформ є те, що їх компоненти мало сумісні з іншими платформами, тому доцільно розглядати їх використання автономно, для відкритих платформ кожний із вище перерахованих компонентів можна використовувати окремо.

Основні переваги та недоліки робототехнічних платформ LEGO Mindstorms та Arduino, представлені в табл. 1.

Таблиця 1

	LEGO	Arduino
Переваги	<ul style="list-style-type: none"> - найбільш розповсюджена конструктор роботів; - велика методична база на російській та українській мовах; - від користувача вимагається не дуже велика кваліфікація; - конструктор дуже міцний; 	<ul style="list-style-type: none"> - найбільш розповсюджена платформа для дорослої робототехніки та електроніки; - платформа повністю відкрита; - велика сумісність з будь-якими іншими модулями; - від користувача вимагається більш висока кваліфікація;
Недоліки	<ul style="list-style-type: none"> - для серйозних завдань не призначений; - закритий не сумісний з будь-якими іншими модулями; 	<ul style="list-style-type: none"> - порівняна ламкість; - плата керування з невеликим рівнем захисту від неправильних підключень;

	- програмувати можна або у візуальних середовищах або на C++.	- програмувати можна або у візуальних середовищах або на C++.
--	---	---

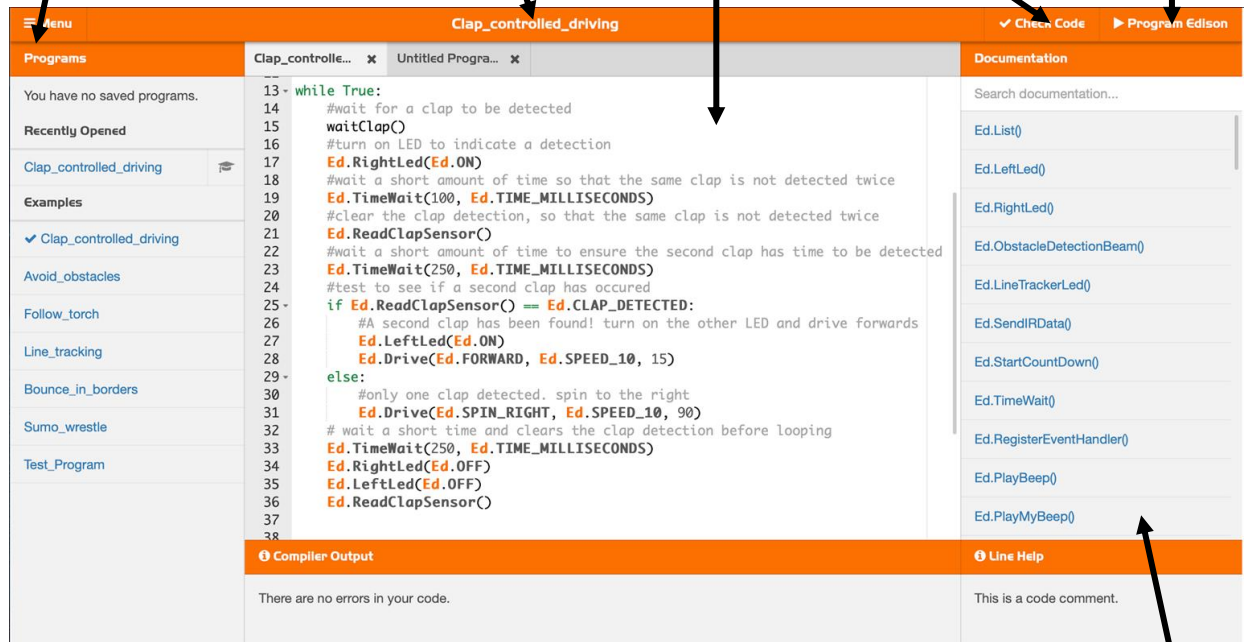
EdPy – це надзвичайно універсальна текстова мова програмування для робота Edison. EdPy (рис. 1.17) базується на Python, популярній мові комп’ютерного програмування, яка відома своєю простотою у вивченні та високою читабельністю. Програмісти використовують Python для розробки всього: від найновіших програм до програмного забезпечення для бізнесу. Facebook, YouTube і Pinterest – лише деякі з багатьох програм, які використовують Python. EdPy робить програмування цікавим, дозволяючи студентам побачити, як їхній код оживає в роботі Edison. За допомогою EdPy студенти вивчають основи справжньої мови програмування та можуть вивести своє дослідження робототехніки та програмування на більш високий рівень. [22]

Меню, щоб відкрити або створити програму

Назва програми

Перевірка коду
Область програмування

Завантажити програму для роботи



Бібліотека

Рис. 1.17. Середовище програмування EdPy

Онлайн-середовище програмування EdPy має низку функцій, розроблених, щоб допомогти навчитися програмуванню. Для роботи у цьому середовищі використовують роботи Edison версії 1 й Edison версії 2.0. Доступ до EdPy можна отримати за адресою www.edpyapp.com. [15]

Робот Edison – приклад мобільного робота (рис. 1.18). Це програмований робот, який має колеса для руху, як автомобіль. Робота Edison можна програмувати за допомогою мови програмування EdScratch або EdPython.



Рис. 1.18. Складові робота Edison

Датчики та кнопки Edison

За замовчуванням кнопки Edison налаштовані таким чином:

Кнопка запису: 1 натискання = завантажити програму,

3 натискання = зчитати штрихкод

Кнопка стоп: 1 натискання = зупинити програму

Кнопка запуску: 1 натискання = запустити програму.



Рис. 1.19. Складові робота Edison

Датчик руху по лінії складається з 2 частин: червоний світлодіодний та світловий сенсор. Також датчик руху по лінії зчитує спеціальні штрих коди, які активують завантажені на нього програми.

Робот Edison має два двигуни, що дають йому можливість рухатися. Він також має кілька датчиків, що дозволяють “бачити” і “відчувати” навколишнє середовище для взаємодії з ним.

Робот Edison допомагає розвивати обчислювальне мислення захопливим та практичним способом. Роботи Edison, розроблені для аудиторних занять, довговічні, не мають зайвих деталей і, завдяки своїм компактним розмірам, їх легко зберігати. Робот працює від самого запуску, вбудований необхідними датчиками, готовими до роботи. Не потрібно встановлювати програмне забезпечення та керувати ним, що полегшує використання Edison разом з різними типами пристроїв та платформ, включаючи ноутбуки, хромбуки та планшети.

Tinkercad – це потужний онлайн-інструмент для 3D-моделювання, який дозволяє користувачам проектувати складні робототехнічні системи без необхідності наявності глибоких знань у САД-програмуванні. Завдяки простому інтерфейсу та широким можливостям, Tinkercad стає популярним серед студентів, інженерів та ентузіастів робототехніки для створення прототипів і макетів роботів.

Tinkercad – це онлайн-платформа для 3D-моделювання, що дозволяє легко створювати та моделювати електронні схеми, друковані плати (PCB), а також проектувати різноманітні 3D-об’єкти. Tinkercad забезпечує інтуїтивно зрозумілий інтерфейс, що робить його доступним навіть для новачків.
[Рис. 1.20]

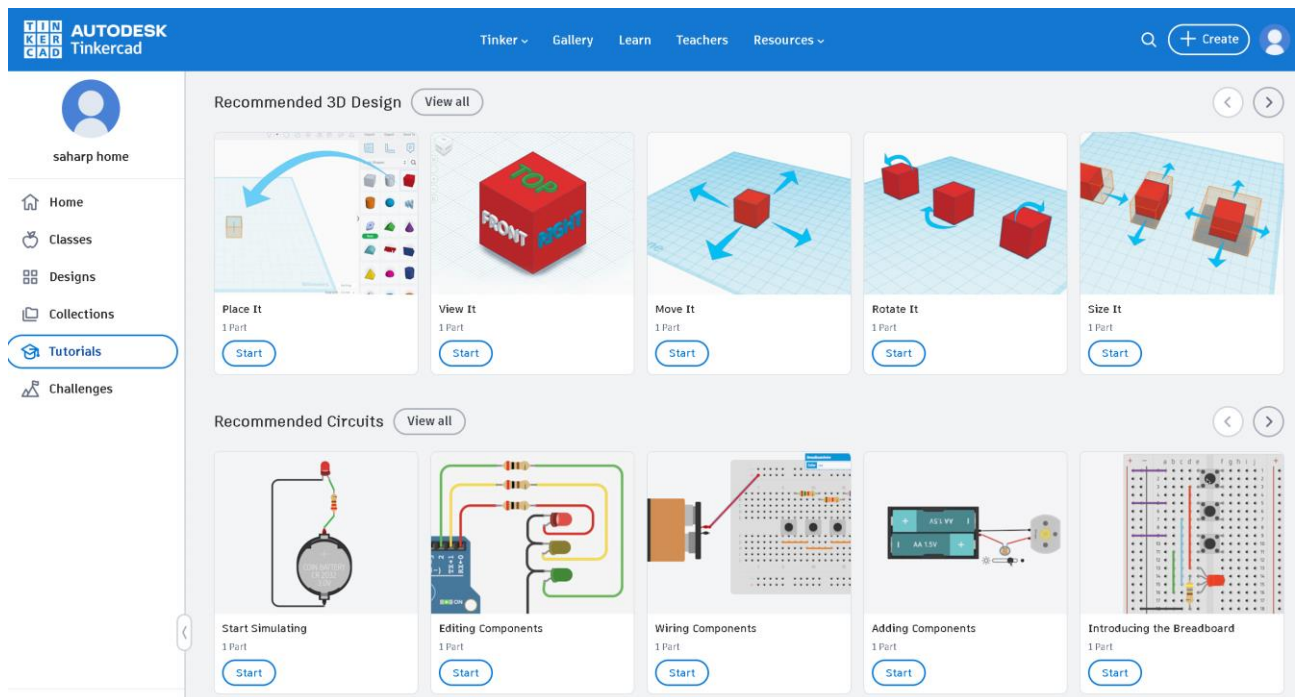


Рис. 1.20. Середовище Tinkercad

Ключові можливості Tinkercad:

- 3D-моделювання: просте створення моделей для 3D-друку або прототипування.
- Електроніка: симуляція електронних схем, підключення різних компонентів (Arduino, датчики, моторчики).
- Програмування: написання коду для Arduino, що дозволяє інтегрувати програмне управління в роботизовані системи.

Проектування складної робототехнічної системи включає кілька ключових етапів, від моделювання механічних частин до створення електронної частини і написання програмного забезпечення. Давайте розглянемо, як цей процес виглядає в Tinkercad.

У Tinkercad можна почати з базового 3D-моделювання для створення корпусу робота. За допомогою простих геометричних фігур (циліндри, куби, прямокутники) можна швидко створити макет майбутнього робота. Важливо враховувати розміри компонування: де будуть розміщені двигуни, датчики та інші елементи. Всі ці деталі повинні бути узгоджені з розмірами, що дозволить компонувати всі елементи у єдиній конструкції.

Використання під час навчання Tinkercad дає можливість створювати деталі, які можуть бути підключені до рухомих частин (наприклад, колеса або ланцюгові механізми). Для цього використовуються функції групування та обертання об'єктів. Можна створити з'єднання для коліс робота або інших рухомих частин, додавши правильні підшипники чи обертальні механізми.

Tinkercad має інструменти для проектування та симуляції електронних схем. Для створення робототехнічних систем потрібно налаштувати правильну схему підключення електронних компонентів.

Tinkercad підтримує симуляцію Arduino. Це дозволяє проектувати схеми, підключати датчики, моторчики, світлодіоди і інші компоненти до Arduino і тестувати їх функціональність без необхідності фізично будувати схему. Можна підключити різні датчики (наприклад, ультразвуковий датчик відстані, датчик температури, датчик руху) до Arduino для отримання необхідних даних.

Для керування робототехнічними системами можна використовувати моторчики і сервоприводи, налаштовуючи їх відповідно до завдань (наприклад, керування рухом робота чи маніпуляторів).

В Tinkercad доступні стандартні компоненти для робототехніки, зокрема різні типи мотори, сенсори (наприклад, для вимірювання відстані, температури чи освітленості), перемикачі та акумулятори. Вибір правильних компонентів важливий для проектування складної системи. Наприклад, для роботів, що мають автономне рухання, використовують двигуни з вбудованими колесами та моторні драйвери для керування швидкістю і напрямком.

Важливо планувати, де будуть розміщуватися компоненти у фізичному просторі робота. У Tinkercad ви можете візуалізувати і симулювати ці компоненти перед тим, як почати фізичну збірку. Компоненти потрібно правильно підключати між собою, використовуючи проводи (на панелі в Tinkercad вони є різних кольорів для зручності). Наприклад, підключення двигунів через моторні драйвери до Arduino.

Однією з головних переваг Tinkercad є можливість програмування роботів безпосередньо на платформі. В Tinkercad доступний онлайн-редактор для програмування на Arduino, який підтримує мову C++. Ви можете написати код для керування моторчиками, датчиками та іншими елементами. Програмування може включати в себе такі завдання, як: управління рухом робота, відображення даних з датчиків на моніторі, зміна поведінки робота залежно від отриманих даних.

Отже, Tinkercad є потужним інструментом для проектування складних робототехнічних систем. Він дозволяє не тільки створювати 3D-моделі роботів, але й інтегрувати електронні компоненти та програмувати їх за допомогою Arduino. Це робить процес проектування доступним навіть для новачків, що дозволяє швидко створювати і тестувати роботизовані системи.

Завдяки широким можливостям Tinkercad, ви можете реалізувати різноманітні ідеї в робототехніці: від простих маніпуляторів до складних автономних роботів. Це інструмент, який підтримує швидку прототипізацію і дозволяє робити проекти більш ефективними та інноваційними.

Висновки до першого розділу

У процесі огляду робототехнічних систем встановлено, що вони застосовуються в різних сферах суспільного життя: освіті, промисловості, медицині, побуті, транспорті, сільському господарстві та наукових дослідженнях. Це підтверджує актуальність їх вивчення у процесі професійної підготовки фахівців з цифрових технологій. Опанування основ робототехніки дає змогу студентам краще зрозуміти принципи функціонування автоматизованих пристроїв, мікроконтролерів, сенсорів, виконавчих механізмів і програмного керування технічними об'єктами.

Проаналізовано особливості вивчення робототехнічних систем при підготовці майбутніх фахівців з цифрових технологій. Визначено, що

ефективне навчання робототехніки має ґрунтуватися на поєднанні теоретичних знань із практичною діяльністю. Важливе значення має виконання лабораторних і практичних робіт, створення простих робототехнічних моделей, робота з мікроконтролерами, використання датчиків, програмування алгоритмів керування та аналіз результатів роботи системи. Такий підхід сприяє розвитку професійної компетентності, самостійності, творчості та готовності студентів до використання цифрових технологій у майбутній діяльності.

Окрему увагу приділено огляду програмних середовищ для вивчення робототехнічних систем. Установлено, що сучасні цифрові платформи дають змогу моделювати електронні схеми, програмувати мікроконтролери, перевіряти роботу пристроїв у віртуальному середовищі та зменшувати потребу у великій кількості фізичного обладнання на початкових етапах навчання. Такі середовища є зручним інструментом для організації навчального процесу, оскільки забезпечують наочність, доступність, інтерактивність і можливість багаторазового тестування створених моделей.

Отже, теоретичний аналіз засвідчив, що підготовка майбутніх фахівців з цифрових технологій до застосування робототехнічних систем є важливим напрямом сучасної професійної освіти. Вивчення робототехніки сприяє формуванню цифрової, технічної, проєктної та дослідницької компетентностей студентів. Подальше дослідження доцільно спрямувати на розроблення методики вивчення робототехнічних систем у цифровому середовищі та перевірку її ефективності в освітньому процесі.

РОЗДІЛ II. МЕТОДИКА ВИВЧЕННЯ РОБОТОТЕХНІЧНИХ СИСТЕМ В ПРОЦЕСІ ПІДГОТОВКИ МАЙБУТНІХ ФАХІВЦІВ З ЦИФРОВИХ ТЕХНОЛОГІЙ

2.1. Вивчення робототехнічних систем на гурткових заняттях

Підготовка майбутніх фахівців з цифрових технологій до застосування робототехнічних систем є важливим складником сучасної професійної освіти. У зв'язку з активним розвитком цифровізації, автоматизації та впровадженням інтелектуальних технічних систем у різні сфери діяльності зростає потреба у фахівцях, які здатні не лише користуватися готовими цифровими інструментами, а й розуміти принципи їх функціонування, проєктувати, моделювати та програмувати робототехнічні пристрої.

У процесі такої підготовки важливе значення має формування в здобувачів освіти системи знань про будову робототехнічних систем, принципи роботи мікроконтролерів, датчиків, виконавчих механізмів, електронних компонентів і програмного забезпечення. Особлива увага приділяється розвитку практичних умінь створення електричних схем, написання алгоритмів, програмування пристроїв, аналізу результатів роботи системи та усунення можливих помилок.

Ефективна підготовка майбутніх фахівців з цифрових технологій передбачає поєднання теоретичного навчання з практичною діяльністю. Саме тому доцільним є використання цифрових середовищ моделювання, зокрема Tinkercad, яке дає змогу створювати віртуальні робототехнічні системи, перевіряти їхню роботу в режимі симуляції та поступово переходити від простих завдань до складніших проєктів. Такий підхід забезпечує наочність навчання, підвищує пізнавальну активність студентів і сприяє формуванню професійних компетентностей.

На базі Тернопільського національного педагогічного університету імені В. Гнатюка для бакалаврів спеціальності Професійна освіта

(Комп'ютерні технології) було започатковано гурткові заняття з робототехніки під назвою “Майстерня Лідер”. Було розроблено програму факультативного гуртка.

Метою програми є навчання основам алгоритмізації та програмування з використанням роботів UNO Robot, розвиток науково-технічного та творчого потенціалу здобувачів освіти шляхом організації їх діяльності в процесі інтеграції початкового інженерно-технічного конструювання та основ робототехніки.

Завдання програми є:

- розвиток навичок проєктування та збирання моделей роботів, а також навички створення алгоритмів та їх програмна реалізація;
- розвиток вмінь працювати в різних програмних середовищах;
- формування наукового світогляду та підтримувати стійку мотивацію до навчання та вивчення нового;
- розвиток студента, зосереджуючи увагу на розвитку логічного мислення, інформаційної культури, пам'яті, уваги та інтуїції.

Навчальна програма гуртка робототехнічні системи розрахована на два семестри навчання (70 годин) з розрахунку 2 години на тиждень заняття складаються: 2 модулів для 1 модуля – 5 тем та для 2.

Програма курсу побудована за принципом «від простого до складного» і охоплює наступні аспекти робототехніки:

- Типові та нестандартні інженерні прийоми у конструюванні роботів;
- Алгоритми для вирішення різноманітних робототехнічних задач та їх реалізація у програмному засобі Tinkercad;
- Фізичні принципи дії датчиків та їх використання;
- Створення власних алгоритмічних програм.

Характеристика змісту навчання

1 семестр. Проєктування робототехнічних систем

Тема 1. Історія розвитку робототехніки

Тема 2. Давачі та їх призначення. Ультразвуковий давач відстані до об'єкта.

Тема 3. Сервопривід в Arduino. Принцип роботи.

Тема 4. Давачі вологості та температури, призначення та принцип роботи.

Тема 5. Давач кислотності водяної суміші. Принцип роботи та програмування давача.

2 семестр. Використання мови програмування для розробки робототехнічних систем

Тема 6. Розробка проекту робототехнічної системи на основі давача лінії.

Тема 7. Керування джойстиком та використання кнопок у проектах Ардуіно.

Тема 8. Давачі сили та згину

Тема 9. Проектування складних робототехнічних систем у Tinkercad

Тема 10. Використання світлодіодних стрічок у Tinkercad

Метою використання програми факультативного заняття є засвоєння студентами системи теоретичних знань із робототехніки, можливість надати студентам системні знання з робототехніки та програмування, сформувані вміння і навички, що необхідні для раціонального використання сучасних інформаційних технологій у процесі моделювання та виготовлення робіт.

Основні завдання полягають у формуванні в студентів таких компетентностей:

– *пізнавальної*: засвоєння початкових технічних і технологічних знань, елементарних уявлень і понять, ознайомлення зі світом техніки, найпростішими технологічними процесами, елементарною електротехнікою, технічним моделюванням, конструюванням і дизайном ознайомлення із новітніми інформаційними технологіями;

– *практичної*: формування графічної грамотності, вмінь і навичок роботи з різноманітними матеріалами та інструментами, виготовлення моделей робіт і механізмів, вміння застосовувати отримані знання на практиці;

- *творчої*: набуття досвіду власної творчої діяльності; розвиток здібностей, просторового й логічного мислення, уяви, фантазії, здатності проявляти творчу ініціативу, вирішувати творчі завдання; формування стійкого інтересу

до технічної творчості, потреби у творчій самореалізації та духовному самовдосконаленні;

– *соціальної*: виховання поваги до праці та людей праці, дбайливого ставлення до навколишнього середовища, культури праці, формування позитивних якостей емоційно-вольової сфери (самостійність, наполегливість, працелюбність та ін.), доброзичливості й товарищкості, уміння працювати в колективі.

Ключова та предметна ІКТ-компетентності навчального курсу

Вивчення пропедевтичного курсу «Робототехніки» сприяє формуванню і розвитку у здобувачів вищої освіти *ключових компетентностей*, серед яких можна виділити предметну креативність, комунікабельність, критичне мислення, науково-технічну грамотність. [12]

Навчання робототехніки спрямоване, перш за все, на розвиток

- креативності;
- комунікативних навичок;
- навичок роботи в команді;
- навичок роботи над проектами.

Основним навчальним принципом робототехніки є принцип «Навчання через дію». Навчання через дію відбувається тоді, коли здобувач освіти створює реальні речі в матеріальному світі і одночасно набуває знань.

Устаткування:

На групу – 6 навчальних комплектів, із розрахунку 1 комплект на 2-х студентів. Навчальний комплект складається із базового набору: UNO Robot. До кожного комплекта потрібен комп'ютер з програмним забезпеченням Tinkercad, що вільно завантажується з офіційного сайту.

Методичний комплект складається з пакету матеріалів до кожного заняття:

- Розгорнута презентація з коментарями;
- Робочі листи для студентів;
- Методичні рекомендації для викладача;

- Інструкції по збиранню робота приклади та вказівки для творчого проекту.

Для наочного прикладу розробки методики проведення занять з даної теми, пропонуємо провести два факультативних заняття: заняття засвоєння нових знань, та заняття застосування знань, умінь та навичок (творча робота).

Перевірка та оцінювання знань й умінь студентів здійснюється під час виконання студентами практичних робіт, а також у формі вікторин, змагань, під час захисту наукових робіт та науково-дослідницьких і прикладних проектів або у формі підсумкових, залікових занять, опитувань, участі в технічних конкурсах, виставках, змаганнях.

Для підтвердження ефективності проведення факультативних занять для студентів першого курсу щодо формування в них наукового світогляду та розвиток навичок проєктування та програмування, розглянемо методику проведення кількох занять нашого гуртка: заняття засвоєння нових знань, та заняття застосування знань, умінь та навичок (творча робота).

Заняття засвоєння нових знань складається з:

- Актуалізації і корекції опорних знань;
- повідомлення студентам теми, мети і завдань заняття;
- мотивації навчання студентів;
- сприймання й усвідомлення студентами фактичного матеріалу, осмислення зв'язків і залежностей між елементами виучуваного;
- узагальнення та систематизація знань;
- рефлексії;
- підсумків заняття.

Заняття з застосування знань, умінь та навичок складається з:

- пригадування отриманих знань з попередніх занять, актуалізації та корекції опорних знань, умінь і навичок;
- повідомлення теми, мети й завдань заняття і мотивації навчання студентів;
- осмислення змісту послідовності застосування способів виконання дій;

- самостійного виконання студентами завдань під контролем і з допомогою викладача;
- звіту студентів про роботу й теоретичне обґрунтування отриманих результатів;
- підсумків заняття.

На творчих роботах передбачається самостійне виконання студентами певних завдань. Мета їх проведення – перевірка і оцінювання навичок та вмінь студентів, що передбачає оцінювання вміння застосовувати набуті знання при самостійній роботі. При проведенні творчих/практичних робіт мають бути розроблені спеціальні інструкції, в яких були б викладені: мета роботи, перелік необхідних знань та вмінь, стислий теоретичний матеріал, приклад виконання завдання з поясненням виконання кожного окремого кроку, індивідуальні завдання, запитання для самоконтролю та вимоги щодо звітності виконання роботи.

Отже, підготовка майбутніх фахівців з цифрових технологій до застосування робототехнічних систем має бути спрямована на розвиток технічного, алгоритмічного та проєктного мислення, умінь працювати з цифровими інструментами, аналізувати технічні завдання й знаходити ефективні способи їх розв'язання. Використання робототехнічних систем у навчальному процесі створює умови для формування практичного досвіду, необхідного для подальшої професійної діяльності в галузі цифрових технологій.

2.2. Методика вивчення робототехнічних систем в цифровому середовищі Tinkercad

Для демонстрації методики розглянемо лабораторну роботу на тему «Підключення світлодіодів до Arduino Uno. Робота з мікроконтролером» у середовищі програмування Tinkercad та лабораторну роботу на тему: «Особливості підключення та керування сервоприводом у середовищі Tinkercad».

Продемонструємо на прикладі структури проведення лабораторного заняття *засвоєння нових знань*, починаємо заняття з:

1 етапу – самовизначення до діяльності (організаційний момент) він займає 1 хв, метод і прийом діяльності студентів і викладача – словесний, зміст етапу – організація групи і вітання з студентами.

2 етап – постановка цілей і завдань, загальна мотивація (підготовка студентів до активного засвоєння нового матеріалу, займає 3 хвилини, використовується проблемний метод для створення проблемної ситуації та пошуку її рішення, форма навчальної діяльності на даному етапі – фронтальна.

3 етап – первинне подання даних студентам (пояснення нового матеріалу, основних понять), займає 15 хвилин, використовується пояснювально-ілюстративний метод, для подання та роз'яснення нового матеріалу студентам, форма навчальної діяльності на даному етапі – фронтальна.

4 етап – застосування, займає 50-55 хвилин, закріплення матеріалу на основі застосування набутих знань на практиці, вироблення відповідних умінь та навичок, форма та метод роботи – робота з комп'ютером (інструкціями).

5 етап – систематизація й узагальнення, займає 10-15 хвилин, поглиблення і осмислення студентами навчального матеріалу, перевірка розуміння сутності нових понять, метод і прийом роботи – робота з робочими листами студентів. проведення дослідів і експериментів.

6 етап – рефлексія (підсумок заняття), займає 5 хвилини, метод роботи – бесіда, підбиття підсумків заняття, бесіда з студентами, вербальне оцінювання робіт студентів, форма навчальної діяльності на даному етапі – колективна.

Згідно з розробленої програми, пропонуємо розглянути лабораторну роботу на тему «Підключення світлодіодів до Arduino Uno. Робота з мікроконтролером» у середовищі програмування Tinkercad

Лабораторна робота

Тема. Підключення світлодіодів до Arduino Uno. Робота з мікроконтролером.

Мета. ознайомитися із способом роботи платформи Tinkercad, дослідити основні програмні модулі та функції для підключення світлодіодів до Arduino Uno за допомогою платформи Tinkercad.

Обладнання і програмне забезпечення:

- персональний комп'ютер (ПК);
- програмне забезпечення: операційна система Windows 7,10;
- онлайн середовище розробки Tinkercad

Теоретичні відомості

Тінкеркад (Tinkercad Circuits Arduino) – безкоштовний, дуже простий і одночасно потужний емулятор Arduino, з якого можна починати навчання електроніці та робототехніці. Він надає дуже зручне середовище для написання своїх проектів. Не потрібно нічого купувати, нічого качати - все доступно онлайн.

Мікроконтролер. це логічне пристрій, який створено для управління іншими пристроями за допомогою логічних (цифрових) сигналів. Це означає, що максимум можна зняти з порта 40 мА, а рекомендується не більше 20 мА. Що станеться, якщо зняти з порта більше, ніж він може віддати? Він зламається. Що буде, якщо зняти з декількох портів більше, ніж може віддати мікроконтролер в цілому? Згорить мікроконтролер. Тому нічого потужніше світлодіода та маленькою пищалки до мікроконтролера підключати не можна. Ніяких моторчиків, лампочок, нагрівачів, потужних радіо-модулів та іншого

живити від цифрових портів не можна. Цифрові порти служать для подачі команд іншим пристроям, наприклад реле / транзисторів для комутації навантажень. Цифровий порт може перебувати в двох станах, вхід та вихід. Режим роботи вибирається за допомогою функції `pinMode (pin, mode)`, де `pin` це номер порта, а `mode` це режим роботи (INPUT. вхід, OUTPUT. вихід, INPUT_PULLUP.вхід підтягнутий до живлення (PushUp)). За замовчуванням всі порти налаштовані як входи (INPUT).

Для зміни режимів роботи портів D2-D13, A0 -A5 можна використати фрагмент програми: `for (byte i = 2; i <= 19; i ++) { pinMode (i, OUTPUT) ; // робимо виходами } 3 93` Для формування цифрового сигналу використовується функція `digitalWrite (pin, value)`, де `pin` це цифровий порт Arduino, підписаний на платі як D; `value`. рівень сигналу: HIGH високий, LOW низький. Також можна використовувати цифри 0 та 1. Приклад, в якому порти ініціалізуються як виходи, і на них подається сигнал: `void setup () { pinMode (10, OUTPUT) ; // D10 як вихід pinMode (A3, OUTPUT) ; // A3 як вихід pinMode (19, OUTPUT) ; // A5 як вихід (Nano / UNO) digitalWrite (10, HIGH) ; // високий рівень на D10 digitalWrite (A3, 1) ; // високий рівень на A3 digitalWrite (19, 1) ; // високий рівень на A5 } void loop () {}` Порт, налаштований як OUTPUT, за замовчуванням має сигнал LOW. Цифровий порт може «вимірювати» напругу, але повідомити він може тільки про її відсутність (сигнал низького рівня, LOW) або наявність (сигнал високого рівня, HIGH). Відсутність напруги вважається проміжок від 0 до 2,1V. Відповідно від 2.1V до VCC (до 5V) мікроконтролер вважає за наявність сигналу високого рівня. Таким чином мікроконтролер може працювати з логічними пристроями, які шлють йому високий сигнал з напругою 3.3V, він такий сигнал прийме як HIGH. Не можна подавати на цифровий порт напругу вище напруги живлення мікроконтролера. Для читання рівня сигналу на порту використовується функція `digitalRead (pin)`, де `pin` це номер порта на платі Arduino. Вони підписані як D , а також порти A0-A5. Функція повертає 0, якщо сигнал низького рівня, і 1 якщо сигнал високого рівня.

Приклад:

```
void setup ()  
{  
  Serial. begin ( 9600 ) ;  
}  
void loop ()  
{  
  Serial. println ( digitalRead ( 5 ))  
; }
```

Завдання на виконання:

1. Створити модель з трьома мигаючими світлодіодами на платформі Tinkercad
2. Додати малу макетну плату;
3. Задати діапазон включення світлодіода 2 сек.

Форма звіту:

1. Тема, мета роботи.
2. Послідовність виконання роботи (опис+скріншоти)
3. Висновок (коротко).
4. Результат виконання без макетної плати (Рис. 2.1)

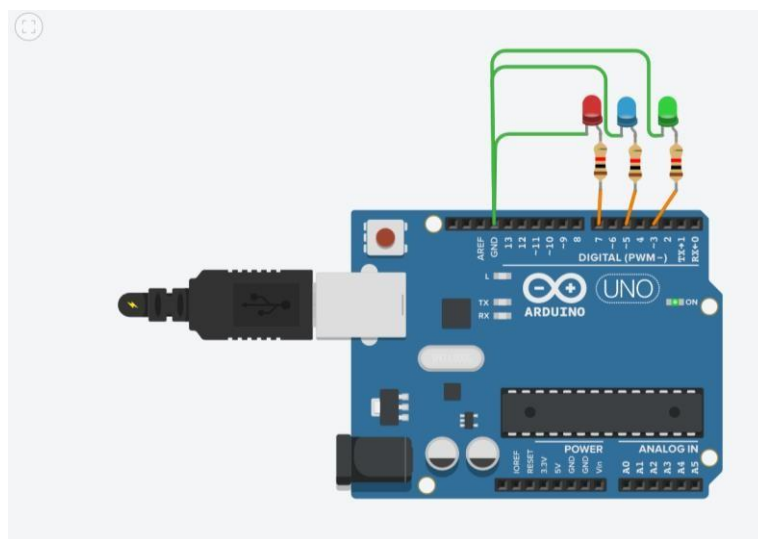


Рис.2.1. Схема підключення

Контрольні питання

1. Зазначте основні сфери застосування Arduino?
2. Які є основні функції цифрового введення та виведення? 3. Які значення має параметр value функції digitalWrite ()?
4. Чи повертає значення функція pinMode (pin, mode)?
5. Описати алгоритм ініціалізації порту вводу-виводу Arduino?
6. Чому в лабораторній роботі світлодіоди підключені без використання транзисторів?
7. Що виконує інструкція ++pin?
8. У чому різниця між змінними int та unsigned int? 9. Що повертає функція millis()?

Лабораторна робота

Тема. Особливості підключення та керування сервоприводом у середовищі Tinkercad.

Мета. Ознайомитись з поняттям мікросервопривід та його функціональними особливостями. Визначити особливості підключення сервоприводу.

Обладнання і програмне забезпечення:

- персональний комп'ютер (ПК);
- програмне забезпечення: операційна система Windows 7,10; - онлайн середовище розробки Tinkercad.

Теоретичні відомості

Сервопривід – це пристрій, що призначений для переміщення у визначених межах регулюючого елемента в системах автоматичного керування або системах дистанційного керування та ін. Сервоприводом можна вважати любий тип механічного приводу (пристрою, органу), що має в своєму складі датчик (положення, швидкості, зусилля та ін.) та блок керування приводом (електронна схема або механічна система тяг), які автоматично підтримують необхідні параметри, відповідно до початкового заданого

значення (положення ручки керування, чисельне значення від інших систем). Простіше кажучи, сервопривід є автоматичним точним виконувачем – отримуючи на вхід значення керуючого параметра (в режимі реального часу) на основі вимірювань датчика, намагається створити та підтримати це значення на виході виконуючого механізму. Зазвичай до контролера сервопривід підключається за допомогою 3 проводів (поз 1) - коричневий або чорний – земля (мінус); - червоний - плюс джерела живлення; - білий або оранжевий – сигнал керування; На рисунку 2.2 приведена узагальнена схема сервопривода.

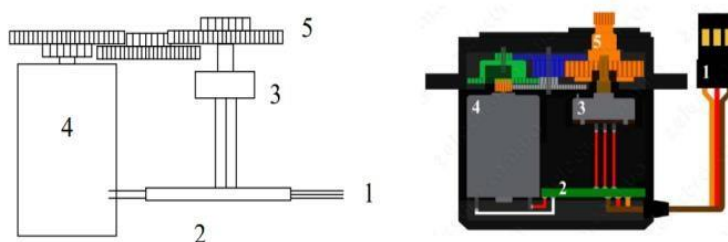


Рис. 2.2. Узагальнена схема сервопривода

Сигнал поступає на плату (поз 2), яка і буде перетворювати даний сигнал і імпульси, що керують електричним двигуном (поз 4). Завдяки потенціометру (поз 3) можна визначати і встановлювати кут повороту сервоприводу. Обертаючи движок потенціометра, відбувається збільшення опору, а разом з тим зменшується напруга, що буде зніматись з центрального виводу. 4 3 2 1 5
Движок потенціометра з'єднаний з вихідним валом сервопривода, відповідно під час обертання вала ми змінюємо значення опору потенціометра. Мікродвигун (поз 4) не здатен розвинути потужне зусилля на валу (момент), але має високу швидкість обертання. Для перетворення високої кутової швидкості з низьким моментом обертання в низьку кутову швидкість з високим моментом обертання використовується пристрій, що має назву редуктор. Редуктор представляє собою набір декількох шестерень, що з'єднують електродвигун та вихідний вал сервоприводу. Шестерня з меншою кількістю зубців веде шестерню з більшою кількістю зубців. Через це швидкість обертання зменшується, а момент обертання збільшується.

Мультиплікатор – це пристрій, що здійснює перетворення низької швидкості обертання вала з високим моментом в високу швидкість обертання з низьким моментом. Тобто цей прилад виконує протилежні редуктору функції.

Завдання до виконання лабораторної роботи:

1. Створити модель з мікросервоприводу (сигнал) на платформі Tinkercad.
2. Додати до побудованої моделі кнопку та задати для неї, команду обертання сервоприводу під кутом 20 градусів.
3. Додати до побудованої моделі кнопку та задати для неї, команду обертання сервоприводу під кутом 40 градусів.

Форма звіту:

1. Тема, мета роботи.
2. Послідовність виконання роботи (опис+скріншоти)
3. Висновок (коротко).
4. Результат виконання роботи (Рис.2.3)

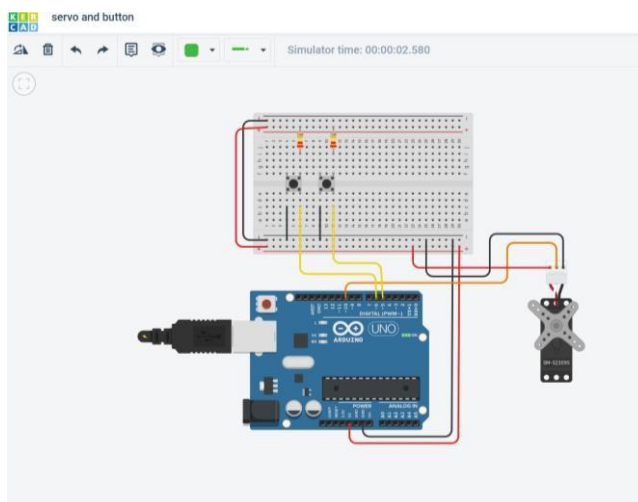


Рис. 2.3. Результат виконання роботи

Контрольні запитання:

1. Що таке сервопривід і які різновиди сервоприводів існують?
2. За якими ознаками відрізняються сервоприводи?
3. Як відбувається керування сервоприводом керуючими сигналами?
4. До яких виводів підключається сервопривід до платформи Arduino?

5. Назвіть основні команди Arduino для керування сервоприводом?

Отже, використання цифрового середовища Tinkercad у процесі вивчення робототехнічних систем дає змогу ефективно поєднати теоретичні знання з практичною діяльністю. На прикладі лабораторних робіт із підключення світлодіодів до Arduino Uno та керування сервоприводом студенти засвоюють основні принципи роботи мікроконтролера, особливості складання електричних схем, написання програмного коду та перевірки працездатності створених моделей у віртуальному середовищі. Такий підхід сприяє формуванню практичних умінь із робототехніки, розвитку алгоритмічного й технічного мислення, а також підвищує безпеку й доступність навчання, оскільки дозволяє моделювати роботу пристроїв без використання фізичних компонентів на початковому етапі. Таким чином, Tinkercad є доцільним інструментом для організації лабораторних занять із робототехніки та підготовки здобувачів освіти до подальшої роботи з реальними робототехнічними системами.

2.3. Апробація розробленої методики в процесі підготовки фахівців з цифрових технологій

На другому етапі нашої роботи ми провели формувальний експеримент, де метою нашої роботи було розроблено конспект лекції [Додаток А, Б] та лабораторних занять з використанням цифрового середовища Tinkercad у процесі вивчення робототехнічних систем, та зробити перевірку за допомогою експерименту ефективності їх використання в освітньому процесі. З контрольною групою ми проводили традиційне навчання за наявним планом викладача, а в експериментальній групі, яка відвідувала позааудиторні заняття на яких використовували запропоновану нами методику навчання. Основні моменти використання освітніх матеріалів були описані в пункті 2.2.

Апробація розробленої методики вивчення робототехнічних систем у цифровому середовищі Tinkercad здійснювалася в процесі підготовки фахівців з цифрових технологій з метою перевірки її ефективності, практичної доцільності та можливостей використання у навчальному процесі. Основна увага приділялася формуванню в здобувачів освіти практичних умінь проектування.

У межах апробації було використано лабораторні роботи, спрямовані на ознайомлення студентів із принципами підключення електронних компонентів, створення електричних схем, написання програмного коду та тестування роботи пристроїв у віртуальному середовищі. Зокрема, розглядалися завдання з підключення світлодіодів до Arduino Uno, роботи з мікроконтролером, а також особливості підключення та керування сервоприводом у середовищі Tinkercad.

Виконання лабораторних робіт дало змогу здобувачам освіти послідовно пройти всі етапи роботи з робототехнічною системою: від аналізу схеми та вибору необхідних компонентів до створення моделі, програмування її функціонування й перевірки результатів симуляції. Такий підхід сприяв кращому засвоєнню теоретичного матеріалу, розвитку алгоритмічного мислення, технічної грамотності, навичок розв'язання практичних завдань і самостійної навчальної діяльності.

Третім етапом нашого дослідження було проведення контрольного тестування, що полягало у діагностиці рівня знань студентів.

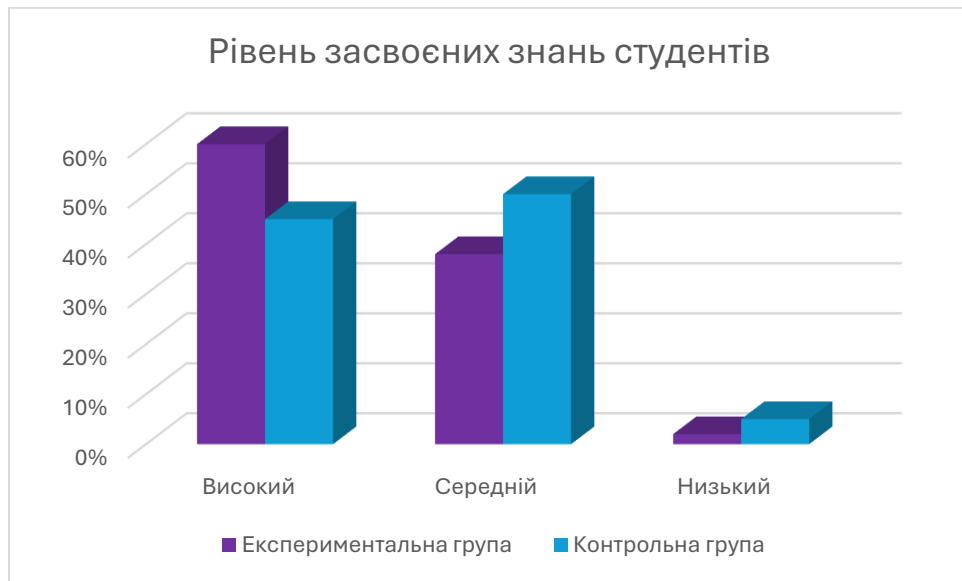
Результати дослідження рівня засвоєння знань із застосуванням розробленої методики навчання наведено в таблиці 2.

Таблиця 2.

Результати контрольного анкетування

№ з/п	Рівень засвоєних знань студентів	Експериментальна група	Контрольна група
1.	Високий	55%	45%
2.	Середній	40%	45%

3.	Низький	5%	10%
----	---------	----	-----



Діаграма 1. Рівень засвоєних знань студентів

Результати апробації засвідчили, що використання цифрового середовища Tinkercad підвищує ефективність вивчення основ робототехніки, оскільки забезпечує наочність навчального матеріалу, інтерактивність, безпечне моделювання електронних схем та можливість багаторазового тестування створених проєктів без ризику пошкодження фізичних компонентів. Це особливо важливо на початковому етапі підготовки фахівців з цифрових технологій, коли студенти лише формують базові вміння роботи з мікроконтролерами та електронними пристроями.

Отже, апробація розробленої методики підтвердила її доцільність для використання в освітньому процесі підготовки фахівців з цифрових технологій. Застосування Tinkercad сприяє формуванню професійно значущих компетентностей, поєднує теоретичну підготовку з практичним моделюванням і створює умови для подальшого поглибленого вивчення робототехнічних систем у реальному апаратному середовищі.

Висновки до другого розділу

У другому розділі було розкрито методику вивчення робототехнічних систем у процесі підготовки майбутніх фахівців з цифрових технологій. Основну увагу зосереджено на практичних підходах до організації навчальної діяльності, що сприяють формуванню в студентів умінь працювати з робототехнічними системами, програмувати мікроконтролери, створювати електронні схеми та аналізувати результати виконання практичних завдань.

Розглянуто можливості вивчення робототехнічних систем на гурткових заняттях. Встановлено, що така форма організації освітньої діяльності має значний навчальний потенціал, оскільки дає змогу поглибити знання студентів, розвивати їхню технічну творчість, самостійність, дослідницькі вміння та інтерес до цифрових технологій. Гурткова робота створює сприятливі умови для виконання практичних завдань, розроблення власних проєктів, моделювання робототехнічних пристроїв і застосування здобутих знань у прикладних ситуаціях.

Окрему увагу приділено методиці вивчення робототехнічних систем у цифровому середовищі Tinkercad. З'ясовано, що це середовище є ефективним інструментом для ознайомлення студентів з основами електроніки, програмування та робототехніки. Використання Tinkercad дає можливість створювати й тестувати електронні схеми у віртуальному середовищі, працювати з Arduino Uno, світлодіодами, резисторами, сервоприводами та іншими компонентами без потреби у постійному використанні фізичного обладнання. Це особливо важливо на початкових етапах навчання, коли студенти лише формують базові практичні навички.

У процесі розроблення методики було визначено доцільність поєднання теоретичного пояснення, демонстрації, покрокового виконання лабораторних робіт, самостійної практичної діяльності та аналізу отриманих результатів. Такий підхід забезпечує поступове формування в майбутніх фахівців з цифрових технологій умінь проєктувати прості робототехнічні системи,

складати електронні схеми, писати програмний код, виявляти помилки та коригувати роботу створених моделей.

Апробація розробленої методики засвідчила її практичну доцільність у процесі підготовки фахівців з цифрових технологій. Виконання лабораторних робіт у середовищі Tinkercad сприяло підвищенню зацікавленості студентів у вивченні робототехніки, активізації їхньої пізнавальної діяльності, розвитку логічного та алгоритмічного мислення, а також формуванню практичних навичок роботи з робототехнічними системами.

Отже, результати другого розділу підтверджують, що ефективне вивчення робототехнічних систем має ґрунтуватися на практико-орієнтованому підході, використанні цифрових середовищ моделювання та залученні студентів до активної навчальної діяльності. Застосування середовища Tinkercad у поєднанні з гуртковими заняттями та лабораторними роботами є доцільним засобом підготовки майбутніх фахівців з цифрових технологій до використання робототехнічних систем у професійній діяльності.

ЗАГАЛЬНІ ВИСНОВКИ

У процесі виконання дослідження було розглянуто проблему підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем. Актуальність обраної теми зумовлена тим, що робототехніка є одним із важливих напрямів розвитку сучасних цифрових технологій, оскільки поєднує знання з програмування, електроніки, моделювання, автоматизації та проєктування технічних систем. У зв'язку з цим підготовка фахівців, здатних працювати з робототехнічними системами, є важливим завданням сучасної освіти.

Відповідно до об'єкта дослідження було проаналізовано процес підготовки майбутніх фахівців з цифрових технологій, а предметом дослідження визначено методику вивчення робототехнічних систем на прикладі цифрового середовища Tinkercad. У ході дослідження було досягнуто поставленої мети – розроблено методику підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем та здійснено перевірку її ефективності в освітньому процесі.

1. Проаналізовано теоретичні засади підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем. З'ясовано, що робототехнічні системи є складними технічними об'єктами, які функціонують на основі взаємодії апаратного та програмного забезпечення. Їх вивчення сприяє формуванню в студентів цифрової, технічної, алгоритмічної, проєктної та дослідницької компетентностей. Встановлено, що ефективна підготовка майбутніх фахівців має ґрунтуватися на поєднанні теоретичних знань із практичною діяльністю, що дає змогу студентам не лише засвоювати навчальний матеріал, а й застосовувати його для розв'язання конкретних прикладних завдань.

2. Розроблено методику вивчення робототехнічних систем у процесі підготовки майбутніх фахівців з цифрових технологій. Визначено, що доцільними формами організації навчання є гурткові заняття, лабораторні роботи, практичні завдання, проєктна діяльність і робота в цифрових

середовища моделювання. Особливу увагу приділено використанню середовища Tinkercad, яке дає змогу моделювати електронні схеми, працювати з Arduino Uno, програмувати мікроконтролери, перевіряти правильність підключення компонентів і тестувати роботу створених моделей у віртуальному просторі. Це забезпечує доступність, наочність та інтерактивність навчального процесу.

3. Експериментальна перевірка запропонованої методики підтвердила її ефективність у процесі підготовки майбутніх фахівців з цифрових технологій. Використання середовища Tinkercad дало змогу покращити рівень зацікавленості студентів у вивченні робототехніки, підвищити розуміння принципів роботи мікроконтролерів, сенсорів і виконавчих механізмів, а також сформувані практичні навички проєктування та програмування простих робототехнічних систем.

Отже, поставлені завдання дослідження було виконано: проаналізовано теоретичні засади підготовки майбутніх фахівців з цифрових технологій до застосування робототехнічних систем; розроблено методику вивчення робототехнічних систем у процесі їхньої професійної підготовки; експериментально перевірено ефективність запропонованої методики в освітньому процесі. Результати дослідження підтверджують, що використання цифрового середовища Tinkercad є доцільним та ефективним засобом формування готовності майбутніх фахівців з цифрових технологій до застосування робототехнічних систем у навчальній і подальшій професійній діяльності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. An Ancient Greek Computer. Доступ до ресурсу: <http://etl.uom.gr/mr/Antikythera/price.htm>
2. Biography of Blaise Pascal, 17th Century Inventor of the Calculator. Доступ до ресурсу: <http://inventors.about.com/library/inventors/blpascal.htm>
3. Brenton O'Brien, Kat Kennewell. EdPy worksheets: Student worksheets and activity sheets. Доступ до ресурсу: <https://meetedison.com/content/EdPy-student-worksheets-complete.pdf> С. 1-96
4. EdPy programming language. Доступ до ресурсу: <https://meetedison.com/robot-programming-software/edpy/>
5. EdPy worksheets. Доступ до ресурсу: <https://meetedison.com/content/EdPy-student-worksheets-complete.pdf>
6. Educator story robot Доступ до ресурсу: http://www.space.gc.ca/asc/pdf/educator-story_robot.pdf
7. History of robots and robotics. Доступ до ресурсу <https://robotnik.eu/history-of-robots-and-robotics/>
8. Robotics: A Brief History. Доступ до ресурсу <https://cs.stanford.edu/people/eroberts/courses/soco/projects/1998-99/robotics/history.html>
9. RoboTuna I (robot-tuna). URL: <http://www.robotic-fish.net/index.php?lang=en&id=robots>
10. Samuel Morland Доступ до ресурсу: http://en.wikipedia.org/wiki/Samuel_Morland
11. Talos. Доступ до ресурсу: <http://en.wikipedia.org/wiki/Talos> .
12. Бабенко В. В. Моделювання та аналіз кінематики мобільних роботів // Механізація та автоматизація виробництва. - 2015. - № 1(47). - С. 43-47.
13. Биков В.Ю. Моделі організаційних систем відкритої освіти: монографія. Київ: Атіка, 2008. 684 с.
14. Волинський П. В., Ящик О. Б., Основи навчання робототехніки у загальноосвітній школі / Актуальні проблеми та перспективи

- технологічної і професійної освіти // Матеріали VII всеукраїнської науково-практичної інтернет-конференції – ТНПУ ім. В. Гнатюка, 20-21 квітня 2023 р. – 17-22 с.
15. Глузман О.В. Базові компетентності: сутність та значення в життєвому успіху особистості. Педагогіка і психологія. 2009. № 2. С. 51-61.
 16. Головка О. В., Беліков В. В., Терехова Т. В. Статистичне моделювання динаміки руху робота з використанням хмарних технологій // Системні технології. - 2019. - № 1(85). - С. 98-106.
 17. Довідник термінів та понять з інформаційно-технічних засобів навчання. Навчальний посібник / укладачі Рак В. І., Луцик І. Б., Потапчук О. І., Франко Ю. П. Ящик О. Б. – Тернопіль, ТНПУ, 2021. – 192 с.
 18. Козіброда С. В., Пальчик А. О., Мазур І.-С. В., Рак В. І. Модель системи віддаленого навчання програмуванню роботів. Серія: Педагогіка. Тернопіль : ТНПУ ім. В. Гнатюка, 2022. №3. URL: <http://nzp.tnpu.edu.ua/issue/archive>.
 19. Коробейніков С. Ю., Герман О. О. Моделювання та аналіз роботизованих комплексів: навчальний посібник / Київський політехнічний інститут. - Київ: КПІ, 2013. - 352 с.
 20. Лавриненко Д. В. Алгоритми визначення та контролю руху робототехнічних систем: монографія / Д. В. Лавриненко, В. М. Коваленко, Д. В. Лавриненко (молодший). - К.: НТУУ "КПІ", 2019. - 192 с. 12
 21. Лук'янов С. Є., Білоус М. В. Дослідження процесу управління повітряним роботом на основі системи візуальної навігації з використанням алгоритму відбору ознак // Вісник Національного технічного університету України «Київський політехнічний інститут». Серія Радіотехніка. Радіоапаратобудування. - 2016. - № 66. - С. 69-74.
 22. Мазур І.-С. В. Використання платформи Tinkercad Circuits у процесі вивчення основ робототехніки майбутніми інженерами-педагогами / І.-С. В. Мазур, Ю. П. Франко // Сучасні цифрові технології та інноваційні

- методики навчання: досвід, тенденції, перспективи : матеріали XI Міжнародної науковопрактичної інтернет-конференції (м. Тернопіль, 6 квітня, 2023). – Тернопіль : ТНПУ ім. В. Гнатюка, 2023. – С. 201-204.
23. Мазур, І.-С., Франко, Ю. Про необхідність вивчення робототехніки як засобу формування професійних компетентностей майбутніх інженерів-педагогів. Освіта. Інноватика. Практика, 2023. №11(1), с. 16–21. URL: <https://doi.org/10.31110/2616-650X-vol11i1-003>
24. Мокренко П. В., Ядловська В. В. Огляд розвитку робототехніки. Частина I (Робототехніка до XX століття). 2020.
25. Павлюк Е.І., Мокренко П.В., Юриш С.Ю. Моделювання і керування рухом автономного мобільного робота в частково невідомому оточенні. // Вимірювальна техніка та метрологія. С.69-77.
26. Сайт для роботи із серещовищем EdPy. Доступ до ресурсу: <https://www.edpyapp.com/>
27. Стен Вейт. Pre-IBM PC Computers. Доступ до ресурсу: <https://www.pc-history.org/>
28. Струтинська О.В. Зарубіжний досвід навчання освітньої робототехніки. Фізико-математична освіта. 2019. Випуск 3(21). С. 140-149.
29. Управління культури Департаменту культури, молоді та спорту. Зі світу техніки. Доступ до ресурсу: <https://kultura-poltava.gov.ua/novyny/zi-svitu-tehniki/>

ДОДАТКИ

Додаток А

Лекція: Проектування складних робототехнічних систем у Tinkercad

Tinkercad – це потужний онлайн-інструмент для 3D-моделювання, який дозволяє користувачам проектувати складні робототехнічні системи без необхідності наявності глибоких знань у САД-програмуванні. Завдяки простому інтерфейсу та широким можливостям, Tinkercad стає популярним серед студентів, інженерів та ентузіастів робототехніки для створення прототипів і макетів роботів.

У цій лекції ми розглянемо основи проектування складних робототехнічних систем за допомогою Tinkercad, а також як цей інструмент можна використовувати для втілення роботизованих ідей у реальність, зокрема через 3D-моделювання, використання Arduino, датчиків і програмування.

Знайомство з Tinkercad

Tinkercad – це онлайн-платформа для 3D-моделювання, що дозволяє легко створювати та моделювати електронні схеми, друковані плати (PCB), а також проектувати різноманітні 3D-об'єкти. Tinkercad забезпечує інтуїтивно зрозумілий інтерфейс, що робить його доступним навіть для новачків.

Ключові можливості Tinkercad:

- 3D-моделювання: Просте створення моделей для 3D-друку або прототипування.
- Електроніка: Симуляція електронних схем, підключення різних компонентів (Arduino, датчики, моторчики).
- Програмування: Написання коду для Arduino, що дозволяє інтегрувати програмне управління в роботизовані системи.

2. Проектування складних робототехнічних систем

Проектування складної робототехнічної системи включає кілька ключових етапів, від моделювання механічних частин до створення електронної частини і написання програмного забезпечення. Давайте розглянемо, як цей процес виглядає в Tinkercad.

2.1. Моделювання механічної частини

1. Проектування корпусу робота:

- У Tinkercad можна почати з базового 3D-моделювання для створення корпусу робота. За допомогою простих геометричних фігур (циліндри, куби, прямокутники) можна швидко створити макет майбутнього робота.
- Важливо враховувати розміри компонування: де будуть розміщені двигуни, датчики та інші елементи. Всі ці деталі повинні бути узгоджені з розмірами, що дозволить компонувати всі елементи у єдиній конструкції.

2. Рухомі частини:

- Tinkercad дозволяє створювати деталі, які можуть бути підключені до рухомих частин (наприклад, колеса або ланцюгові механізми).
- Для цього використовуються функції групування та обертання об'єктів. Можна створити з'єднання для коліс робота або інших рухомих частин, додавши правильні підшипники чи обертальні механізми.

2.2. Електронні компоненти

Tinkercad має інструменти для проектування та симуляції електронних схем. Для створення робототехнічних систем потрібно налаштувати правильну схему підключення електронних компонентів.

1. Arduino в Tinkercad:

- Tinkercad підтримує симуляцію Arduino. Це дозволяє проектувати схеми, підключати датчики, моторчики, світлодіоди і інші компоненти до Arduino і тестувати їх функціональність без необхідності фізично будувати схему.
- Ви можете підключити різні датчики (наприклад, ультразвуковий датчик відстані, датчик температури, датчик руху) до Arduino для отримання необхідних даних.
- Для керування робототехнічними системами можна використовувати моторчики і сервоприводи, налаштовуючи їх відповідно до завдань (наприклад, керування рухом робота чи маніпуляторів).

2. Підключення датчиків і моторів:

- В Tinkercad доступні стандартні компоненти для робототехніки, зокрема різні типи мотори, сенсори (наприклад, для вимірювання відстані, температури чи освітленості), перемикачі та акумулятори.
- Вибір правильних компонентів важливий для проектування складної системи. Наприклад, для роботів, що мають автономне рухання, використовують двигуни з вбудованими колесами та моторні драйвери для керування швидкістю і напрямком.

3. Розміщення компонентів:

- Важливо планувати, де будуть розміщуватися компоненти у фізичному просторі робота. У Tinkercad ви можете візуалізувати і симулювати ці компоненти перед тим, як почати фізичну збірку.
- Компоненти потрібно правильно підключати між собою, використовуючи проводи (на панелі в Tinkercad вони є різних кольорів для зручності). Наприклад, підключення двигунів через моторні драйвери до Arduino.

2.3. Програмування робота

Однією з головних переваг Tinkercad є можливість програмування роботів безпосередньо на платформі.

1. Програмування на Arduino:

- В Tinkercad доступний онлайн-редактор для програмування на Arduino, який підтримує мову C++. Ви можете написати код для керування моторчиками, датчиками та іншими елементами.
- Програмування може включати в себе такі завдання, як: управління рухом робота, відображення даних з датчиків на моніторі, зміна поведінки робота залежно від отриманих даних.

2. Приклад коду:

- Якщо ви хочете, щоб робот рухався, реагуючи на відстань, можна використати ультразвуковий датчик для вимірювання відстані і відповідно змінювати напрямок або швидкість руху. Ось приклад простого коду для такої задачі:

```

cpp Копировать #include <Servo.h> int trigPin = 9; // Пін для
ультразвукового датчика int echoPin = 10; // Пін для
ультразвукового датчика
Servo motor; // Створюємо об'єкт для сервоприводу

void setup() {  motor.attach(6); // Підключаємо мотор до
пін 6  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  Serial.begin(9600); // Встановлюємо швидкість серійного зв'язку }

void loop() {  long duration,
distance;

  digitalWrite(trigPin, LOW);
delayMicroseconds(2);  digitalWrite(trigPin,
HIGH);  delayMicroseconds(10);
digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);  distance =
duration * 0.034 / 2;

  if (distance < 10) { // Якщо об'єкт знаходиться на відстані менше 10 см  motor.write(0); //
Остановимо мотор
  } else {  motor.write(90); // Запустимо мотор на максимальну швидкість
  }  delay(100);
}

```

Цей код дозволяє роботу реагувати на відстань до об'єкта і регулювати швидкість двигуна в залежності від того, чи є перешкода.

Tinkercad є потужним інструментом для проектування складних робототехнічних систем. Він дозволяє не тільки створювати 3D-моделі роботів, але й інтегрувати електронні компоненти та програмувати їх за допомогою Arduino. Це робить процес проектування доступним навіть для новачків, що дозволяє швидко створювати і тестувати роботизовані системи.

Завдяки широким можливостям Tinkercad, ви можете реалізувати різноманітні ідеї в робототехніці: від простих маніпуляторів до складних автономних роботів. Це інструмент, який підтримує швидку прототипізацію і дозволяє робити проекти більш ефективними та інноваційними.

Лекція: Використання світлодіодних стрічок у Tinkercad

1. Вступ

Світлодіодні стрічки — це популярний елемент у сучасних електронних проектах, особливо в освітлювальних та декоративних системах, а також для індикації стану або передачі інформації в різних пристроях. Завдяки простоті в управлінні та різноманіттю кольорів, світлодіодні стрічки часто використовуються в багатьох робототехнічних, дизайнерських та інтерактивних проектах.

Tinkercad — це чудова платформа для проектування електронних схем, і тут можна не лише моделювати, а й симулювати роботу світлодіодних стрічок. В цій лекції ми розглянемо, як працювати з світлодіодними стрічками в Tinkercad, як налаштувати електричні схеми та програмувати їх для використання в різних проектах.

2. Що таке світлодіодні стрічки?

Світлодіодні стрічки — це гнучкі плати з вбудованими світлодіодами (LED), які зазвичай об'єднані в ланцюг і мають можливість підключення до джерела живлення для створення різних ефектів освітлення. Вони можуть мати різні кольори, яскравість, а деякі навіть змінюють колір залежно від напруги чи сигналу.

В Tinkercad ви можете працювати з різними типами світлодіодних стрічок, включаючи прості однотонні стрічки та RGB-стрічки, які дозволяють отримувати широкий спектр кольорів.

3. Компоненти для роботи зі світлодіодними стрічками в Tinkercad

Tinkercad надає все необхідне для створення проектів зі світлодіодними стрічками: від самих світлодіодів до компонентів для підключення до Arduino для керування ними.

1. **Світлодіоди (LED):** У Tinkercad доступні базові компоненти для симуляції світлодіодів, які можна підключити до схем.
2. **RGB-світлодіоди:** Ці світлодіоди можуть змінювати свій колір за допомогою трьох основних каналів — червоного (R), зеленого (G) і синього (B).
3. **Arduino:** Arduino є основним мікроконтролером, який використовується для керування світлодіодними стрічками. За допомогою Arduino можна налаштувати різноманітні ефекти освітлення та зчитувати сигнали від датчиків, щоб змінювати кольори.
4. **Резистори:** Резистори використовуються для обмеження струму, що проходить через світлодіоди, щоб запобігти їх пошкодженню.

4. Підключення світлодіодної стрічки до Arduino

1. Основи підключення:

- Для початку потрібно розуміти, як правильно підключити світлодіод до мікроконтролера. Якщо ви використовуєте прості світлодіоди, підключіть один кінець до **пін**у на Arduino, а інший — до **землі (GND)** через резистор, щоб обмежити струм.
- Якщо ви працюєте з RGB-світлодіодом, кожен з трьох каналів (R, G, B) підключається до окремого **пін**у Arduino, а загальний контакт світлодіода — до GND.

2. Для світлодіодної стрічки:

- Якщо ви хочете підключити цілу стрічку світлодіодів, вам потрібно використовувати **мотори драйвери** або **спеціальні чіпи** для управління всіма світлодіодами через один або кілька пінів Arduino.
- Також можна використовувати бібліотеку **Adafruit NeoPixel** для роботи з більш складними стрічками, такими як WS2812, яка дозволяє керувати кожним світлодіодом індивідуально.

3. Схема підключення в Tinkercad:

- В Tinkercad ви можете побудувати схему, підключивши світлодіоди до пінів Arduino та додавши резистори для захисту компонентів.
- Ви можете симулювати роботу схеми без необхідності фізичного підключення, перевіряючи, як ваша схема реагує на зміни в програмуванні.

5. Програмування світлодіодних стрічок

Для програмування світлодіодних стрічок в Tinkercad ми використовуємо Arduino та спеціальне програмне забезпечення для управління кольорами і ефектами.

- 1. Програмування простих світлодіодів:** Для простих світлодіодів можна використати базову програму для вмикання та вимикання світлодіода, а також для керування його яскравістю (PWM). Ось приклад коду для світлодіода: cpp

Копировать

```
int ledPin = 13; // Пін, до якого підключений світлодіод
void setup() {
  pinMode(ledPin, OUTPUT); // Встановлюємо пін як вихід
}
void loop() {
  digitalWrite(ledPin, HIGH); // Включаємо світлодіод
  delay(1000); // Затримка 1 секунда
  digitalWrite(ledPin, LOW); // Вимикаємо світлодіод
  delay(1000); // Затримка 1 секунда }

```

Цей код змусить світлодіод мигати раз на секунду.

- 2. Програмування RGB-світлодіодів:** RGB-світлодіоди мають три піни (для червоного, зеленого і синього світла). Використовуючи команду `analogWrite()`, можна налаштувати яскравість кожного каналу і отримувати різні кольори. Ось приклад коду для RGB-світлодіода:

cpp

Копировать

```
int redPin = 9; // Пін для червоного каналу
int greenPin = 10; // Пін для зеленого каналу
int bluePin = 11; // Пін для синього каналу
void setup() {
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
}
void loop() {
  analogWrite(redPin, 255); // Включаємо червоний канал
  analogWrite(greenPin, 0); // Вимикаємо зелений канал
  analogWrite(bluePin, 0); // Вимикаємо синій канал
  delay(1000); // Затримка 1 секунда
  analogWrite(redPin, 0); // Вимикаємо червоний канал
  analogWrite(greenPin, 255); // Включаємо зелений канал
  analogWrite(bluePin, 0); // Вимикаємо синій канал
  delay(1000); // Затримка 1 секунда
}

```

```

analogWrite(redPin, 0); // Вимикаємо червоний канал analogWrite(greenPin, 0); //
Вимикаємо зелений канал analogWrite(bluePin, 255); // Включаємо синій канал
delay(1000); // Затримка 1 секунда
}

```

Цей код дозволяє змінювати колір світлодіода з червоного на зелений і синій по черзі.

3. **Використання бібліотеки Adafruit NeoPixel:** Для більш складних RGB-стрічок, наприклад WS2812, можна використовувати бібліотеку **Adafruit NeoPixel**, яка дозволяє керувати кожним світлодіодом індивідуально.

cpp Копировать

```

#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 8 // Кількість світлодіодів в стрічці
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);
void setup() { strip.begin(); strip.show(); //
Ініціалізація всіх світлодіодів
} void loop() { for(int i=0; i<strip.numPixels(); i++) { strip.setPixelColor(i, strip.Color(255,
0, 0)); // Встановлюємо червоний колір
strip.show(); delay(100);
} for(int i=0; i<strip.numPixels(); i++) { strip.setPixelColor(i, strip.Color(0, 255, 0)); //
Встановлюємо зелений колір strip.show(); delay(100);
}
}

```

Використання світлодіодних стрічок у Tinkercad дозволяє створювати інтерфейси для декоративного освітлення, індикації та інтерактивних проектів, що можуть значно покращити вигляд вашого робота чи іншої електронної системи. Завдяки інтеграції з платформою Arduino, ви маєте можливість програмувати світлодіоди для досягнення різних ефектів освітлення.

Основні переваги роботи з світлодіодними стрічками в Tinkercad:

Простота використання: Tinkercad дозволяє швидко підключити світлодіоди до вашої схеми та симулювати їх роботу. Для простих проектів можна обмежитися базовим підключенням і використовувати базове програмування, а для складніших — підключити RGB-стрічки або навіть використовувати бібліотеки для управління індивідуальними світлодіодами.

Гнучкість управління: Світлодіодні стрічки можуть бути використані для створення різних кольорових ефектів, а також для інтерактивних змін кольору в залежності від умов (наприклад, зміна кольору залежно від показників датчиків або сигналів від інших елементів).

Моделювання і тестування: За допомогою Tinkercad можна легко протестувати свою схему і програму без необхідності фізично збирати систему. Це дозволяє уникнути помилок на етапі проектування і забезпечує швидку перевірку різних варіантів конфігурації світлодіодних стрічок.

Інтерактивність: Завдяки можливості програмувати Arduino для управління світлодіодами, ви можете створювати інтерактивні системи. Наприклад, світлодіоди можуть змінювати свій колір або ефект в залежності від зовнішніх факторів, таких як температура, вологість або рух.

Застосування світлодіодних стрічок у робототехніці та інших проектах:

Освітлення роботів: Світлодіодні стрічки можуть використовуватися для освітлення роботів, надаючи їм ефектний вигляд або навіть допомагаючи в орієнтації в темряві.

Індикатори стану: Ви можете використовувати світлодіоди для індикації стану робота або іншого пристрою (наприклад, червоний для помилок, зелений для готовності до роботи).

Інтерактивні ефекти: Стрічки можуть застосовуватися для створення ефектів освітлення в залежності від взаємодії з користувачем або навколишнім середовищем, наприклад, у розважальних системах або інтерактивних дисплеях.

Індикація помилок: Світлодіоди можуть бути корисні для повідомлення про помилки або особливі стани в робототехнічних проектах, наприклад, миготіння червоного світла може свідчити про аварійний стан системи. Приклад використання світлодіодної стрічки в інтерактивному проекті:

Уявімо проект, де робот оснащений датчиком відстані (ультразвуковий датчик). Світлодіодна стрічка змінює колір залежно від того, наскільки близько знаходиться об'єкт до робота. Якщо відстань менша за 10 см, світлодіоди будуть червоними, якщо відстань між 10 і 20 см — жовтими, а якщо більше — зеленими.

Приклад коду для цього проекту: cpp

Копировать

```
#include <Adafruit_NeoPixel.h>
#define PIN 6
#define NUMPIXELS 8 // Кількість світлодіодів в стрічці
#define trigPin 9 // Пін для ультразвукового датчика
#define echoPin 10 // Пін для ультразвукового датчика
Adafruit_NeoPixel strip(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800); void setup()
{ strip.begin(); strip.show(); pinMode(trigPin, OUTPUT); pinMode(echoPin,
INPUT);
  Serial.begin(9600);
}
void loop() { long duration, distance; digitalWrite(trigPin, LOW);
delayMicroseconds(2); digitalWrite(trigPin, HIGH);
delayMicroseconds(10); digitalWrite(trigPin, LOW); duration =
pulseIn(echoPin, HIGH); distance = duration * 0.034 / 2; if (distance <
10) { // Якщо об'єкт знаходиться дуже близько setStripColor(255, 0,
0); // Червоний колір
}
else if (distance >= 10 && distance <= 20) { // Якщо об'єкт на середній відстані
setStripColor(255, 255, 0); // Жовтий колір
}
else { // Якщо об'єкт знаходиться далеко setStripColor(0, 255, 0);
// Зелений колір
} delay(500);
}
void setStripColor(int red, int green, int blue) { for(int i=0;
i<strip.numPixels(); i++) { strip.setPixelColor(i,
strip.Color(red, green, blue));
} strip.show();
}
```

У цьому проекті світлодіодна стрічка буде змінювати колір залежно від того, як близько знаходиться об'єкт до робота. Це чудовий спосіб продемонструвати використання світлодіодних стрічок у робототехнічних і інтерактивних системах.

Світлодіодні стрічки — це простий, але дуже ефективний спосіб додавання візуальних ефектів до ваших проектів в Tinkercad. Вони дозволяють створювати не лише естетичні ефекти освітлення, але й використовувати їх як індикацію стану, що особливо корисно в робототехнічних системах. Завдяки можливостям симуляції в Tinkercad, ви можете легко тестувати і вивчати роботу світлодіодів, а також інтегрувати їх з іншими електронними компонентами, такими як датчики та Arduino.

Tinkercad: Ваш перший крок у світ 3D-моделювання

Відкрийте для себе потужний, але доступний інструмент для створення тривимірних моделей прямо у браузері — без встановлення додаткового програмного забезпечення.

3D-МОДЕЛЮВАННЯ

ДЛЯ ПОЧАТКІВЦІВ

БЕЗКОШТОВНО



Що таке Tinkercad?

Tinkercad — це безкоштовний браузерний 3D-редактор, розроблений компанією Autodesk. Він створений спеціально для того, щоб зробити 3D-моделювання доступним для кожного — від школярів до досвідчених дизайнерів.

🌐 Браузерний доступ

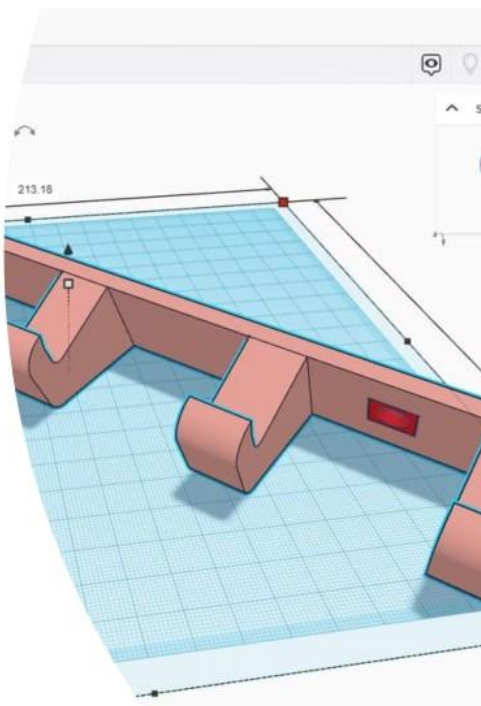
Працює прямо у браузері без встановлення програм. Достатньо мати підключення до інтернету.

🎓 Для всіх рівнів

Ідеально підходить для початківців, учнів, студентів та викладачів будь-якого віку.

⚡ Широкий функціонал

Створюйте 3D-моделі, електронні схеми Arduino та навіть пишть блоки коду для мікроконтролерів.



Початок роботи: Реєстрація та вхід

Як розпочати роботу з Tinkercad?

Реєстрація займає лічені хвилини. Платформа підтримує кілька способів входу, що робить процес максимально зручним.

1. Перейдіть на офіційний сайт www.tinkercad.com
2. Натисніть кнопку «Увійти» або «Зареєструватися»
3. Оберіть зручний спосіб: електронна пошта або обліковий запис Google
4. Підтвердіть реєстрацію та увійдіть до особистого кабінету

Для викладачів

Tinkercad пропонує спеціальний режим для освітян. Викладачі можуть створювати власні класи, запрошувати учнів за спеціальним кодом та відстежувати їхній прогрес у реальному часі.

- 📌 Викладацький акаунт надає доступ до панелі управління класами, спільних проєктів та інструментів оцінювання.

Інтерфейс Tinkercad: Ваша робоча площина

Після входу в систему ви потрапите до головної робочої області. Розуміння кожного елемента інтерфейсу — перший крок до впевненої роботи з платформою.



Робоча площина (Сітка)

Центральна область редактора, де ви розміщуєте та редагуєте всі об'єкти своєї моделі.



Меню проєкту

Верхня панель із доступом до налаштувань, збереження, експорту та загальних параметрів.



Панель інструментів (зліва)

Бібліотека базових форм: куби, циліндри, сфери та інші геометричні примітиви.



Панель властивостей (справа)

Відображає параметри обраного об'єкта: розміри, колір, позицію та інші характеристики.



Навігація

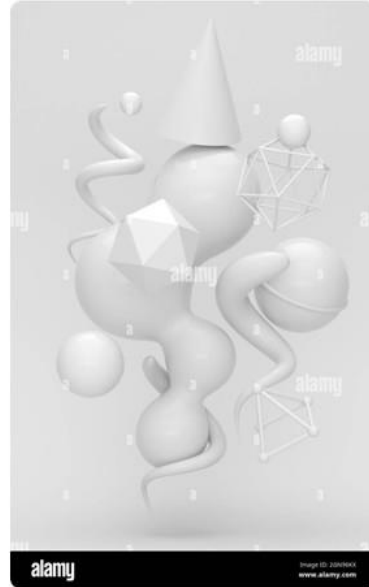
Керування камерою: поворот, масштабування та переміщення сцени за допомогою миші.

Додавання та маніпуляція об'єктами

Як додати об'єкт на сцену?

Робота з об'єктами в Tinkercad інтуїтивно проста. Використовуйте меню **перегляду** (View) для швидкого розміщення форм на робочій площині.

- **Перетягування**
Взьміть потрібну форму з лівої панелі та перетягніть її на сцену.
- **Зміна розміру**
Використовуйте білі маркери на краю об'єкта або введіть точні значення у панелі властивостей.
- **Поворот**
Чорні стрілки навколо об'єкта дозволяють обертати його на 360° по будь-якій осі.



Ключові інструменти для моделювання

Tinkercad надає набір потужних інструментів, які перетворюють прості форми на складні моделі. Ознайомтесь із чотирма основними функціями редактора.



Колір

Змініть колір будь-якого об'єкта одним кліком. Велика палітра дозволяє візуально розрізнити деталі моделі та робити її більш виразною.



Прозорість

Налаштуйте рівень прозорості об'єкта. Корисно для створення ефектів скла, води або для тимчасового приховування деталей під час роботи.



«Дірка» (Hole)

Перетворіть об'єкт на «порожнистий» — він стане інструментом для вирізання отворів у інших формах після групування.



Групування (Group)

Об'єднайте кілька об'єктів в єдину структуру за допомогою **Ctrl+G**. Це спрощує маніпуляції зі складними моделями.

Робота з робочою площиною

Зміна робочої площини

Tinkercad дозволяє переміщати робочу площину на різні рівні, що відкриває можливість для багаторівневого моделювання. Ви можете розміщувати об'єкти не лише на основній сітці, а й на поверхнях інших фігур.

Це особливо корисно при створенні складних конструкцій із кількома ярусами або внутрішніми порожнинами.

Вирівнювання (Align)

Інструмент вирівнювання — один із найважливіших для створення акуратних моделей. Він дозволяє точно розмістити об'єкти відносно один одного.

- Вирівнювання по **лівому / правому краю**
- Вирівнювання по **центру** по горизонталі та вертикалі
- Вирівнювання по **верхньому / нижньому краю**
- Рівномірний **розподіл** об'єктів між собою

Створення складних форм

Комбінуючи прості фігури та використовуючи спеціальні інструменти, ви можете створювати надзвичайно складні та деталізовані моделі.



Комбінювання об'єктів через «Дірку»

Встановіть режим «Дірка» для об'єкта, розмістіть його всередині іншої фігури, а потім застосуйте групування (Ctrl+G). Результат — вирізаний отвір або складна форма з порожнинами.



Групування для об'єднання деталей

Після створення отворів або розміщення всіх елементів виділіть їх і натисніть «Групувати». Всі об'єкти стануть єдиною моделлю, яку можна переміщати та редагувати як ціле.



Інструмент «Скрибл» (Scribble)

Малюйте власні 2D-форми від руки прямо в браузері. Tinkercad автоматично перетворить ваш малюнок на тривимірний об'єкт із можливістю налаштування висоти та товщини.

Збереження та експорт проекту

Автоматичне хмарне збереження

Tinkercad автоматично зберігає ваші проекти в хмарину кожні п'ять секунд після кожної зміни. Ви ніколи не втратите свою роботу — достатньо лише мати підключена до Інтернету.

Всі проекти доступні з будь-якого пристрою після входу в обліковий запис.

Експорт моделей

Готова модель може бути експортована у різних форматах для подальшого використання:

.STL

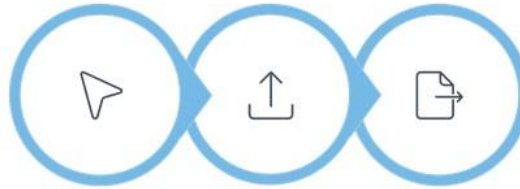
Стандартний формат для 3D-друку

.OBJ

Для використання в інших 3D-програмах

.SVG

2D-вектор для лазерного різання



Оберіть
модель

Натисніть
Export

Виберіть
формат

Процес експорту займає лише кілька кліків — виділіть модель, натисніть кнопку «Експорт» і оберіть потрібний формат.



Ваші перші кроки в 3D-світі!

Експериментуйте. Створюйте. Надихайте.

Tinkercad — це потужний, але водночас простий інструмент, який відкриває двері у світ тривимірного дизайну. Незалежно від того, чи ви хочете надрукувати власну фігурку на 3D-принтері, спроектувати корпус для електронного проекту, чи просто дослідити свої творчі здібності — Tinkercad стане вашим надійним помічником.

🚀 Почніть сьогодні

Зареєструйтеся та спробуйте створити свою першу модель за 15 хвилин.

📚 Навчайтеся

Використовуйте вбудовані уроки Tinkercad та спільноту для натхнення.

🌟 Ділітьсяся

Публікуйте свої роботи та надихайте інших на творчість.



Моделювання та програмування простих робототехнічних систем на базі Arduino Uno

Від першого світлодіода до повноцінної роботизованої руки — практичний курс з електроніки та програмування для початківців.

РОБОТОТЕХНІКА

ARDUINO UNO

ПРОГРАМУВАННЯ

Вступ: Світ роботів починається тут!

Робототехніка — це поєднання механіки, електроніки та програмування, що дозволяє створювати пристрої, які взаємодіють із фізичним світом. Arduino Uno є ідеальною платформою для перших кроків у цьому захопливому напрямку.

Що таке робототехніка?

Наука про проектування, побудову та програмування машин, здатних виконувати завдання автономно або під керуванням людини. Вона охоплює механіку, електроніку та програмне забезпечення.

Чому Arduino Uno?

Простота, доступність та величезна спільнота. Arduino Uno — це відкрита платформа з інтуїтивним середовищем розробки, ідеальна для навчання та прототипування.

Потенціал Arduino

Від простого блимання світлодіодом до складних роботизованих систем із датчиками, двигунами та бездротовим зв'язком — можливості практично безмежні.



Основи Arduino Uno: Серце вашого робота

Мікроконтролер ATmega328P

Серце Arduino Uno — 8-бітний мікроконтролер ATmega328P з тактовою частотою 16 МГц, 32 КБ флеш-пам'яті та 2 КБ оперативної пам'яті SRAM. Він обробляє команди та керує підключеними компонентами.

Цифрові та аналогові піни


Arduino Uno має **14 цифрових пінів** (вхід/вихід) та **6 аналогових входів**. Цифрові піни передають сигнали HIGH/LOW, аналогові — зчитують напругу від 0 до 5 В.

Середовище розробки (IDE)

Arduino IDE — безкоштовна програма для написання, компіляції та завантаження коду на плату. Мова програмування базується на C++ зі спрощеним синтаксисом. Основні функції: `setup()` (ініціалізація) та `loop()` (головний цикл).

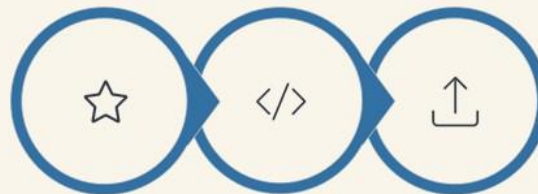
Autodesk Circuits

Онлайн-симулятор, що дозволяє збирати віртуальні схеми та тестувати код без фізичних компонентів. Ідеальний інструмент для навчання та швидкого прототипування перед реальною збіркою.

 **Порада:** Починайте з Autodesk Circuits, щоб уникнути пошкодження компонентів під час перших експериментів.

Перший крок: Просте керування світлодіодом

Класичний "Hello World" у світі електроніки — зупинити світлодіод блимаєти. Цей проект навчає основні підключення компонентів, написання коду та розуміння структури програми Arduino.



Підключення

Написати код

Завантажити

Цей простий проект демонструє ключові концепції програмування мікроконтролерів: ініціалізацію пінів у функції `setup()`, безперервний цикл у `loop()`, а також використання функції `delay()` для керування часовими інтервалами між станами HIGH та LOW.

Схема підключення

- Анод світлодіода → цифровий піл 13
- Катод світлодіода → резистор 220 Ом
- Резистор → GND (земля)

Ключові функції коду

- `pinMode(13, OUTPUT)` — налаштування піна
- `digitalWrite(13, HIGH)` — увімкнути
- `delay(1000)` — затримка 1 секунда
- `digitalWrite(13, LOW)` — вимкнути

Рух — це життя: Керування двигунами

Щоб робот міг рухатися, потрібен драйвер двигунів. Безпосереднє підключення двигуна до Arduino може пошкодити плату через високий струм. Драйвер L298N вирішує цю проблему.

Драйвер L298N: міст між Arduino та двигунами

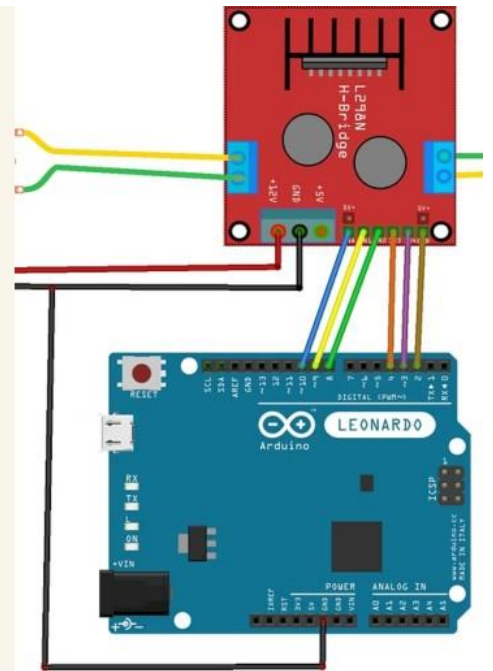
Модуль L298N — це H-міст, що дозволяє керувати напрямком та швидкістю двох DC-двигунів. Він живиться від окремого джерела (5–12 В) і приймає логічні сигнали від Arduino для керування.

Підключення DC-двигунів

Двигуни підключаються до виходів OUT1–OUT4 драйвера. Управлінські входи IN1–IN4 з'єднуються з цифровими пінами Arduino. Піни ENA та ENB керують швидкістю через ШІМ-сигнал (PWM).

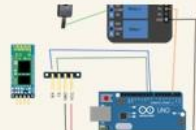
Програмування руху

Комбінація сигналів на IN1/IN2 визначає напрямок: **вперед** (HIGH/LOW), **назад** (LOW/HIGH), **зупинка** (LOW/LOW). Функція `analogWrite()` на ENA/ENB задає швидкість від 0 до 255.



Створення Bluetooth-робомашини: Керування зі смартфона

Об'єднавши Arduino Uno, драйвер L298N, DC-двигуни та Bluetooth-модуль HC-05, можна створити повноцінну керувану робомашину. Керування здійснюється через мобільний застосунок.



Компоненти проекту

Arduino Uno (контролер), HC-05 (Bluetooth-зв'язок), L298N (драйвер двигунів), два DC-двигуни з колесами, шасі та акумулятор 9 В.



Етап конструювання

Збірка шасі, монтаж двигунів, підключення L298N до Arduino та HC-05 до послідовного порту (TX/RX). Важливо дотримуватись правильної полярності живлення.



MIT App Inventor

Візуальне середовище для створення Android-застосунків без написання коду. Кнопки інтерфейсу надсилають символи ('W', 'S', 'A', 'D') через Bluetooth для керування рухом.

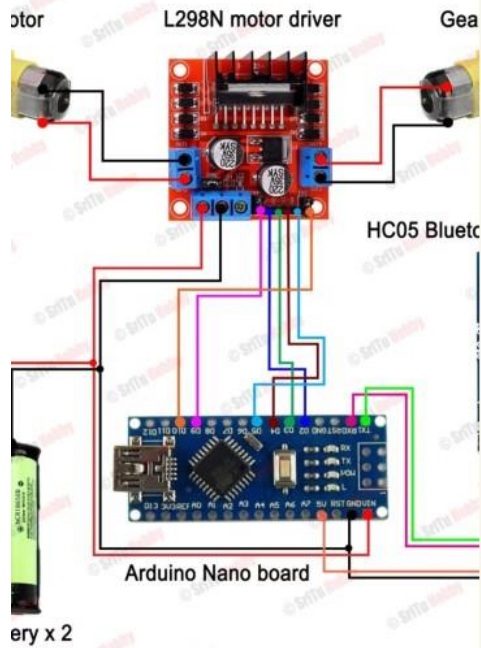


Схема Bluetooth-робомашини

На схемі показано повне підключення всіх компонентів робомашини: Arduino Uno є центральним контролером, HC-05 забезпечує бездротовий зв'язок зі смартфоном, а драйвер L298N керує двома DC-двигунами. Зверніть увагу на правильне підключення живлення: Arduino живиться від USB або зовнішнього джерела 7-12 В, двигуни — від окремого акумулятора через L298N.

📱 Смартфон

MIT App Inventor → Bluetooth команди

📡 HC-05

Прийм даних → передача Arduino через Serial

🔌 L298N

Обробка команд → керування двигунами

🤖 Робот

Рух вперед, назад, повороти

Роботизована рука: Точність та маніпуляції

Сервомотори: точне керування

Сервомотор — це двигун із вбудованим енкадером, що дозволяє точно контролювати кут повороту від 0° до 180°. Arduino керує сервоприводом через ШІМ-сигнал на одному піні. Бібліотека `Servo.h` значно спрощує роботу.

Побудова руки

Конструкція з картону або 3D-друку складається з кількох сегментів (основа, передпліччя, кисть, захоп). Кожен суглоб — окремий сервомотор. Для простого проєкту достатньо 2-4 сервоприводи.

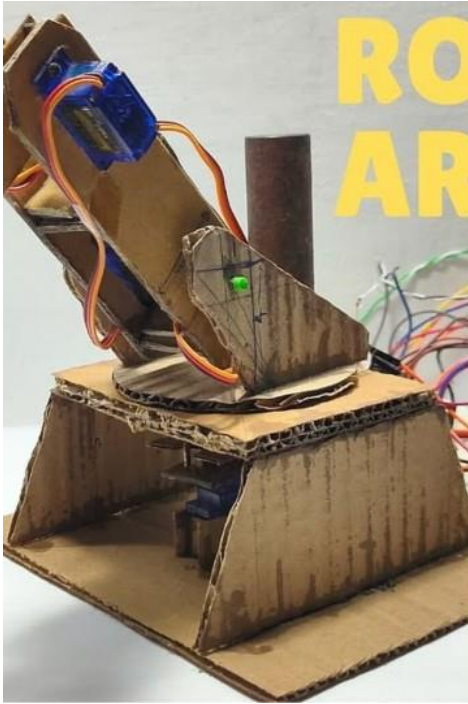
Керування потенціометрами

Потенціометр (змінний резистор) підключається до аналогового входу Arduino. Зчитуючи значення від 0 до 1023, ми перетворюємо його на кут 0-180° і передаємо на сервомотор у реальному часі.

Застосування

- Вибір та переміщення легких об'єктів
- Сортування предметів за кольором або розміром
- Автоматизація простих виробничих операцій
- Навчальні моделі для демонстрації кінематики

📄 Розширення: додайте датчик відстані (ультразвуковий HC-SR04) для автоматичного виявлення об'єктів.



Роботизована рука в дії

Готова роботизована рука на базі Arduino Uno демонструє потужність поєднання простих компонентів та грамотного програмування. Навіть з картону та доступних сервоmotorів можна створити функціональний маніпулятор, здатний виконувати реальні завдання.

4

Сервоmotorи

Оптимальна кількість для повноцінної руки з захопом

180°

Кут повороту

Діапазон керування кожного сервоприводу

20ms

Період ШІМ

Стандартний сигнал керування сервоmotorами

3

Потенціометри

Для ручного керування основними суглобами

Висновки та подальші кроки

Протягом цього курсу ви пройшли шлях від першого світлодіода до створення Bluetooth-керованої роботизованої руки. Ви оволоділи основами електроніки, програмування Arduino та системного мислення.

1

Основи

Світлодіоди, цикли, затримки, цифрові та аналогові сигнали

2

Рух

Драйвер L298N, DC-двигуни, ШІМ-керування швидкістю

3

Зв'язок

Bluetooth HC-05, MIT App Inventor, бездротове керування

4

Маніпуляції

Сервоmotorи, потенціометри, роботизована рука

Що далі?

- Датчики: ультразвукові, інфрачервоні, температури
- Дисплеї: LCD, OLED для відображення даних
- Машинний зір: камери та розпізнавання образів
- Штучний інтелект: TPUML на мікроконтролерах

Ресурси для навчання

- Офіційний сайт arduino.cc — документація та приклади
- [Instructables.com](http://instructables.com) — тисячі проектів спільноти
- YouTube-канал: GreatScott!, Andreas Spiess
- Книга: "Arduino для початківців" (українською)

