

Управління пневматичними приводами через контролер Arduino: підходи до моделювання та візуалізації

Національний аерокосмічний університет «Харківський авіаційний інститут»

В даній роботі розглядаються сучасні підходи до управління пневматичними приводами із використанням мікроконтролера Arduino. Пневматичні приводи широко застосовуються в автоматизованих системах, зокрема у виробничих лініях, мехатроніці та робототехніці, завдяки їх простоті конструкції, швидкості реакції та безпечності. Проте ефективне управління ними вимагає чіткої координації електронної та пневматичної частин системи, що передбачає моделювання динаміки привода, побудову керуючих алгоритмів і зручну візуалізацію процесу. Arduino як відкрита апаратно-програмна платформа є ідеальним інструментом для реалізації прототипів систем керування. У дослідженні проведено аналіз основних типів пневмоприводів (односторонньої та двосторонньої дії), електропневматичних клапанів та датчиків положення. Наведено методику побудови моделі пневматичної системи у середовищі Wokwi та її апаратну реалізацію за допомогою Arduino. Особливу увагу приділено розробці алгоритмів керування на базі ПІ-регуляторів, логіки зворотного зв'язку та часових затримок, які враховують інерційність і стисливість повітря. Також у роботі описано методи візуалізації станів пневмопривода з використанням інтерфейсу FluidSIM demo, а також імітація в середовищі Wokwi, що дозволяє створити інтерактивні системи моніторингу та налагодження. Наведено приклади інтеграції сенсорів тиску, датчиків положення та дисплеїв для зчитування параметрів у реальному часі. Метою дослідження є підвищення ефективності, надійності та зручності управління пневматичними системами на основі недорогих і відкритих платформ. Результати можуть бути корисними для інженерів, студентів і розробників автоматизованих систем, а також знайдуть застосування в навчальних лабораторіях для демонстрації принципів дії пневматичних механізмів та цифрового керування.

Ключові слова: пневматичні приводи, Arduino, Simulink, моделювання, візуалізація, автоматизація, керування приводами, мікроконтролер, сенсори.

Вступ

У сучасному світі автоматизації та цифрових технологій значну роль відіграють виконавчі механізми, серед яких важливе місце займають пневматичні приводи. Вони використовуються в різних галузях – від промисловості до медицини, від транспортних систем до лабораторного обладнання. Їх популярність пояснюється низкою переваг: простою конструкцією, високою швидкістю реагування, екологічністю, доступністю стисненого повітря, а також відносно невисокою вартістю. Завдяки цим властивостям пневматичні системи залишаються затребуваними, попри активний розвиток електроприводів.

Проблема, яка постає перед розробниками таких систем, полягає у забезпеченні ефективного та точного керування пневмоприводами. На відміну від електричних систем, пневматичні виконавчі механізми мають деякі особливості, що ускладнюють побудову систем автоматизації. До них належать стиснення повітря, нелінійна динаміка руху поршня, затримки відклику клапанів та чутливість до зміни навантаження. Тому для ефективного керування такими приводами потрібне точне моделювання фізичних процесів, що відбуваються в

системі, розробка оптимальних алгоритмів управління, а також реалізація засобів візуального контролю над її функціонуванням [1].

З іншого боку, сучасні мікроконтролерні платформи, такі як Arduino, відкривають нові можливості для реалізації адаптивних та гнучких систем керування. Arduino є відкритою апаратно-програмною платформою, що забезпечує простоту у використанні, доступність компонентів і підтримку великої кількості сенсорів та виконавчих елементів. Ця платформа активно використовується у навчальному процесі, для прототипування, а також у промислових і лабораторних умовах як ефективний інструмент для розробки автоматизованих систем [2].

Поєднання Arduino з пневматичними приводами дозволяє створити компактні, економічні та гнучкі системи керування з можливістю зчитування інформації від сенсорів, логічного оброблення сигналів і активації пневматичних елементів відповідно до заданого алгоритму. Для цього необхідно розробити математичні моделі, які дозволяють імітувати роботу пневматичного привода, враховуючи його інерційні та динамічні властивості. Завдяки моделюванню можливо не лише спростити процес проектування, а й попередити помилки під час практичної реалізації.

Ще одним важливим аспектом є візуалізація – процес подання інформації про стан системи у зручному для користувача вигляді. Візуалізація дозволяє моніторити в реальному часі параметри роботи пневматичного привода (тиск, положення поршня, стан клапанів тощо), спрощує діагностику та забезпечує оперативне реагування на відхилення. Реалізація візуальних інтерфейсів можлива як на самому контролері (за допомогою LCD-дисплеїв або LED-індикаторів), так і через ПК, використовуючи такі середовища, як FluidSim, Processing, LabVIEW або MATLAB.

У цій роботі розглядаються сучасні підходи до управління пневматичними приводами із використанням контролера Arduino. Висвітлено методи моделювання пневматичних систем, алгоритми керування рухом приводів, а також реалізація візуалізації станів системи. Наведено приклади використання програмного забезпечення для симуляції, а також апаратної реалізації прототипу керованого привода.

Основна частина

Пневматичні приводи широко використовуються в автоматизованих системах через їхню простоту, надійність та швидкодію. Сучасні тенденції автоматизації передбачають інтеграцію таких приводів із мікроконтролерами, зокрема Arduino, для створення гнучких та програмованих систем керування. Це потребує адекватного теоретичного і практичного підходу до моделювання, керування і візуалізації.

Пневматичні приводи перетворюють енергію стислого повітря у механічний рух (лінійний або обертовий). Основні компоненти можна виділити наступні: виконавчі елементи (пнеumoциліндри (односторонньої та двосторонньої дії) пневмомотори); керуючі елементи (електропневматичні розподільники); інформаційні елементи (датчики положення тощо); елементи логіки (контролер, розподільники) та компресор з системою підготовки повітря [3].

Arduino – відкритий апаратно-програмний мікроконтролер, що підтримує

цифрові/аналогові входи-виходи, ПІД-регулювання, PWM, інтерфейси I2C, UART, SPI. У керуванні пневмоприводами Arduino застосовується для:

- виводу сигналів керування на електропневматичні розподільники;
- зчитування сигналів з датчиків (наприклад, положення поршня);
- реалізації логіки управління (умовні переходи, ПІД-регулятори).

Фізичні процеси в пневмоприводах можна описати диференціальними рівняннями (1), зокрема рівняння:

$$F = P \cdot A = m \cdot a, \quad (1)$$

де P – тиск у камері циліндра, A – площа поршня, m – маса навантаження, a – прискорення.

Це рівняння описує взаємозв'язок між силою, що створюється тиском стисненого повітря, та динамікою руху поршня згідно з другим законом Ньютона.

Враховуються також тертя, сила пружності та витік повітря. Такі моделі дозволяють передбачити динаміку системи та налаштувати ефективне керування.

Для моделювання динаміки і керування використовуються такі середовища:

MATLAB/Simulink – дозволяє моделювати пневмосистему разом із логікою Arduino.

FluidSIM – візуалізація і симуляція пневматичних схем.

Proteus, Tinkercad, Wokwi – для моделювання роботи схеми Arduino з клапанами та датчиками.

Такі симулятори допомагають протестувати алгоритм до створення фізичної установки.

Візуалізація підвищує зручність роботи з системою та дозволяє моніторити стан у реальному часі.

Локальну візуалізацію здійснюють через *дисплейні модулі* (типу LCD 16x2, OLED) застосовуються для візуалізації робочих параметрів: стану виконавчих механізмів (клапанів) і положення поршневих елементів. *LED-індикація* – для швидкої діагностики.

Візуалізація через ПК відбувається за допомогою Serial Monitor / Plotter в Arduino IDE – для виводу графіків та повідомлень.

Processing, Python (PySerial) – створення графічного інтерфейсу з кнопками, індикаторами та графіками.

SCADA/IoT-системи (Node-RED, ThingsBoard тощо) – для візуалізації великих систем через мережу.

Маємо обладнання, яке складається з трьох виконавчих циліндрів. Для прикладу циліндри рухаються в наступній послідовності (2):

$$St-1-n1-2-T1-3-T2-n2-n3 \quad (2)$$

де St – сигнал кнопки старт;

$1, 2, 3$ – циліндри, в висунутому положенні;

$n1, n2, n3$ – циліндри у зворотному крайньому положенні;

$T1, T2$ – таймери (затримка часу між діями).

Arduino отримує сигнал з кнопки запуску. Включає реле розподільника, яке відповідає за висування першого циліндру → подається повітря в порожнину циліндру і поршень починає висуватись. Досягнув крайнього положення спрацьовує датчик положення (або кінцевий вимикач), який фіксує крайнє положення поршня і дає сигнал контролеру Arduino. Це реле розподільника

вимикається і переводить розподільник в положення в якому циліндр починає рухатись в зворотному напрямі. В крайньому положенні спрацьовує кінцевий вимикач і дає сигнал контролеру. Аналогічно відбувається вмикання розподільника для висування другого циліндра. Після спрацювання кінцевого вимикача розподільник продовжує утримувати положення і вмикається таймер $T1$. Після відліку затримки висувається третій циліндр. При досягненні крайнього положення спрацьовує кінцевий вимикач і вмикається таймер $T2$. Після закінчення затримки другий циліндр починає рух у зворотному напрямку, а за ним – третій.

Переваги використання Arduino полягають у простоті та низькій вартості апаратної частини, гнучкості програмування, а також у широких можливостях інтеграції з іншими системами, такими як мобільні застосунки або хмарні сервіси.

Серед недоліків варто відзначити нелінійність пневмоприводів, яка ускладнює точне керування, обмежену обчислювальну потужність Arduino, а також затримки у роботі, спричинені фізичними властивостями пневматики, зокрема стисливістю повітря.

Управління пневматичними приводами через Arduino – ефективне рішення для навчальних, лабораторних та малих виробничих систем. Використання математичного та симуляційного моделювання дозволяє краще розуміти процеси у системі, а засоби візуалізації – контролювати її в реальному часі. Такий підхід забезпечує поєднання практичної гнучкості з теоретичною обґрунтованістю, що робить його актуальним у контексті розвитку сучасних автоматизованих систем.

Даний алгоритм можна зобразити циклограмою (рис. 1) з подальшим написанням рівнянь для запуску кожного реле розподільників. При необхідності на циклограмі можна вказати межі дії кінцевих вимикачів на зовнішній стороні кола або орієнтуватись по внутрішнім дугам, які об'єднують однойменні вершини. Сигнали кінцевих вимикачів позначимо « x + цифра номеру циліндру» - відповідає за висунутий стан і « xn + цифра номеру циліндру» - відповідає за втягнутий стан. Далі ці позначення будемо використовувати в рівняннях.

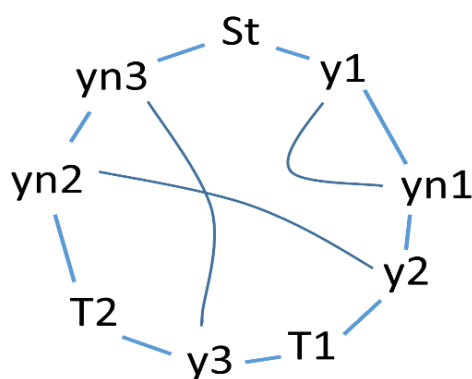


Рис. 1. Циклограма роботи пневматичної системи

Описані сигнали належать до системи керування пневмоприводами (циліндрами), де реле керують розподільниками подачі повітря для переміщення штоків. Нижче наведено опис призначення кожного з сигналів:

$y1$ – керування реле розподільника для висування першого циліндра;

y_2 – керування реле розподільника для висування другого циліндра;
 y_3 – керування реле розподільника для висування третього циліндра;
 $yn1$ – керування реле розподільника для втягування першого циліндра;
 $yn2$ – керування реле розподільника для втягування другого циліндра;
 $yn3$ – керування реле розподільника для втягування третього циліндра;
 y_4 – елемент пам'яті ввімкнути;
 $yn4$ – елемент пам'яті вимкнути;
 $T1$ – запуск таймера 1;
 $T2$ – запуск таймера 2.

На основі прикладу алгоритму далі розпишемо рівняння роботи системи [4].

Оскільки в даній системі використовуються моностабільні розподільники (рис. 2), виникає необхідність утримувати ввімкненим реле весь час, коли циліндр знаходиться у висунутому положенні. Цей момент потрібно враховувати в рівняннях і програмі для контролера додатковими елементами пам'яті.

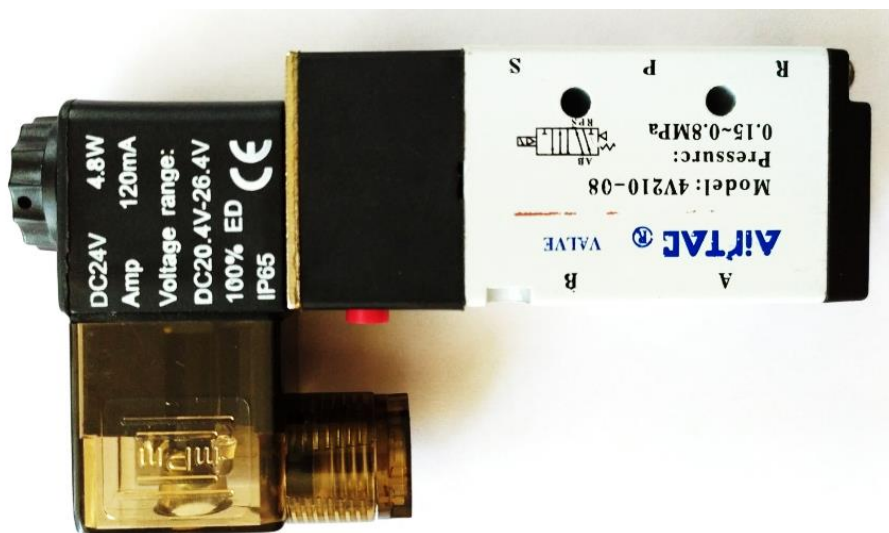


Рис. 2. Розподільник 5/2, моностабільний

Рівняння для вмикання першого розподільника для висування штоку циліндра

$$Y1 = St \& XN2 \& XN3 \& XN4. \quad (3)$$

Оскільки розподільник моностабільний, то його потрібно утримувати ввімкненим до моменту коли потрібно рухатись в зворотньому напрямку, і для цього в нас буде слугувати елемент пам'яті. Це враховуємо в коді. Продовжимо для другого розподільника

$$Y2 = XN1. \quad (4)$$

Утримування другого розподільника будемо здійснювати до отримання сигналу від другого таймеру:

$$T1 = X2,$$

$$Y3 = T1.$$

Утримування третього розподільника здійснюємо до отримання сигналу від кінцевого вимикача другого циліндру у втягнутому стані:

$$T2 = X3.$$

контролер подає команду на реле - рух циліндру, реле формує сигнал відповідного кінцевого вимикача і дає зворотній зв'язок що рух відбувся і імітація досягнення крайнього положення відбулась. Аналогічно симулюється робота другого та третього циліндрів.

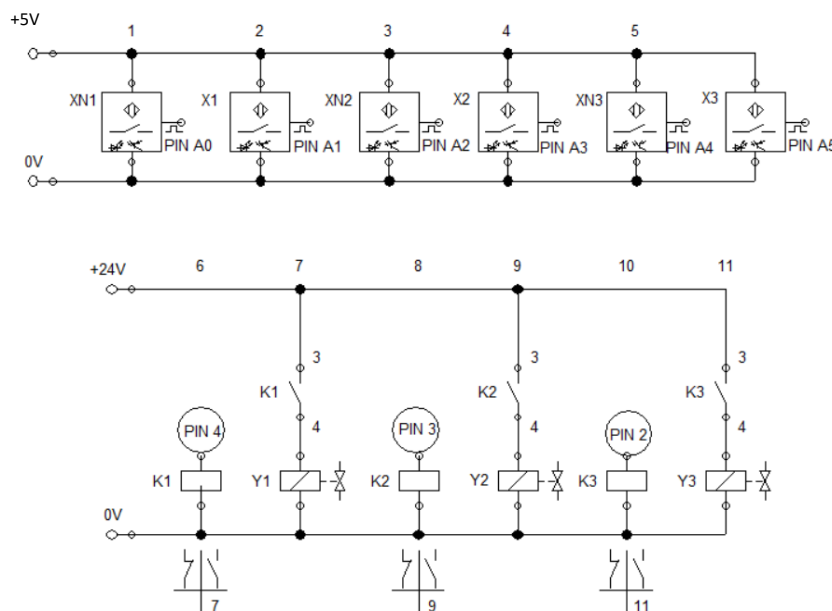


Рис. 4. Схема електрична

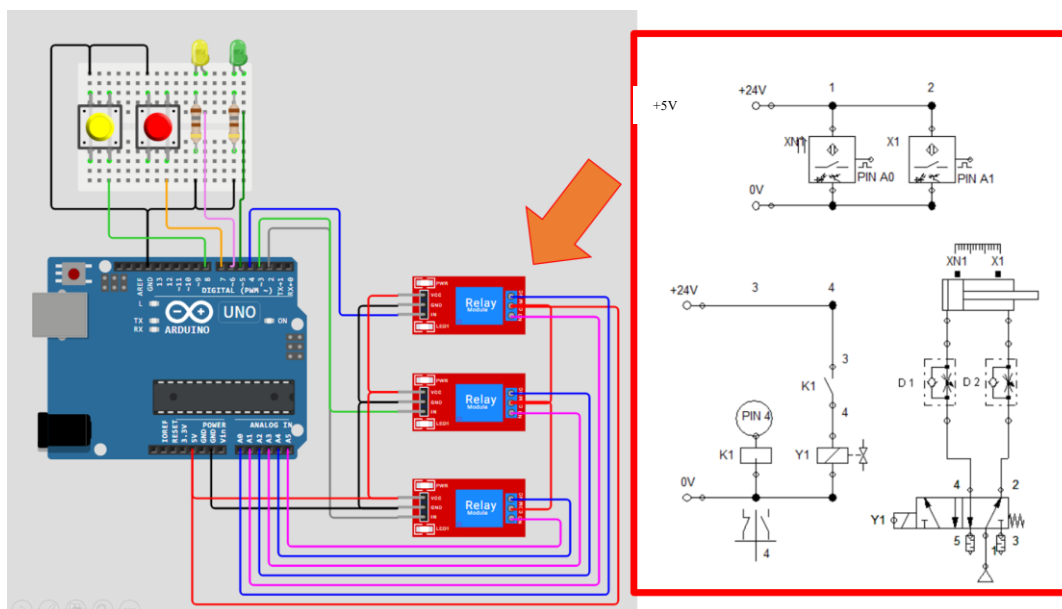


Рис. 5. Схема електрична в симуляторі Wokwi

На основі рівнянь (1-4) розробимо програму для контролера та виконаємо її моделювання (рис. 6).

Після перевірки правильності роботи логіки коду переходимо до стендових

випробувань (рис. 7). На цьому етапі виконується налаштування дроселів (регулювання швидкості руху поршня циліндра), а також, за потреби, коригуються параметри інших елементів.

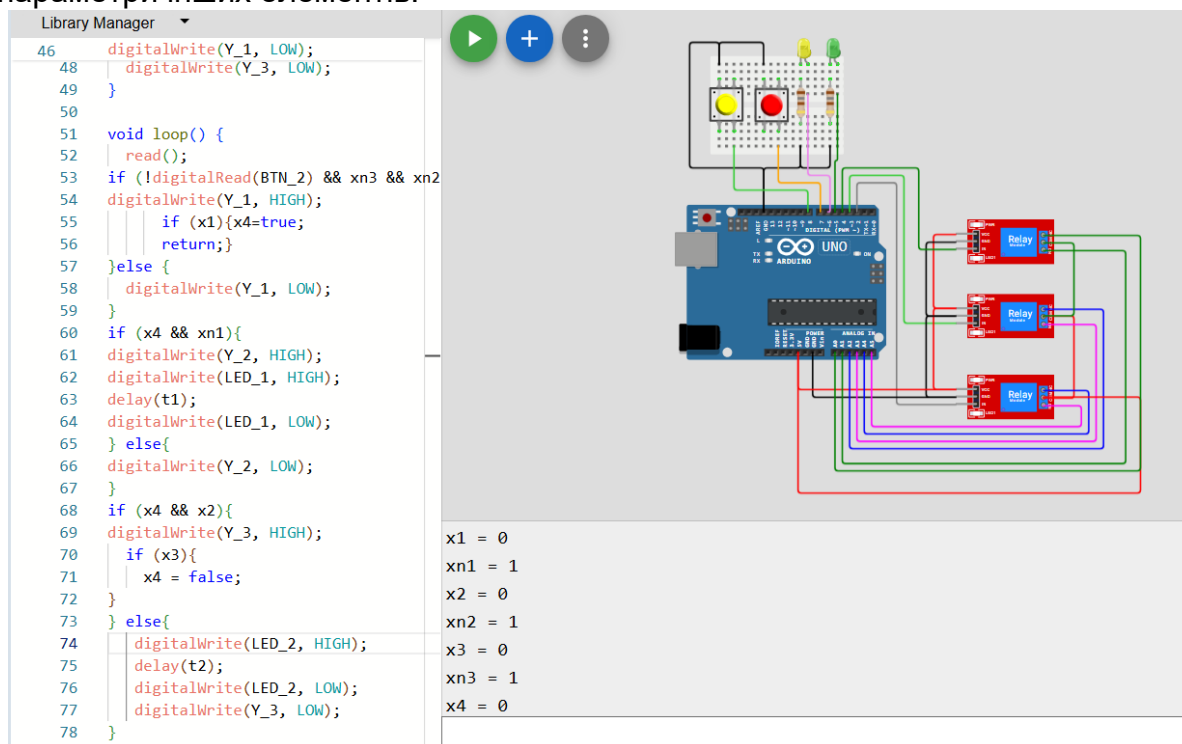


Рис. 6. Симуляція роботи коду в симуляторі Wokwi

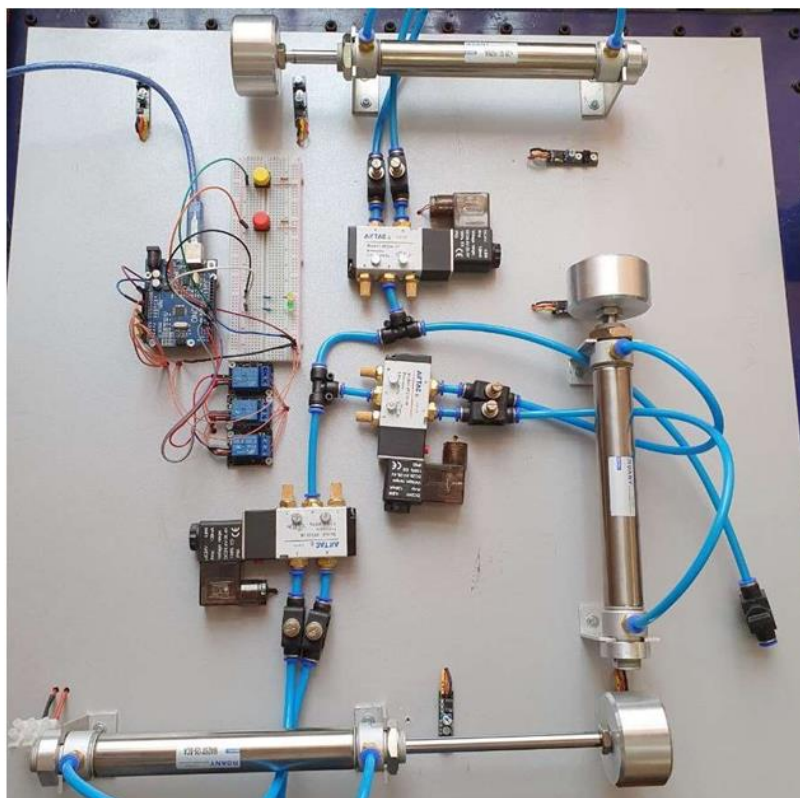


Рис. 7. Пневматичний стенд з керуванням від Arduino Uno

Наступним етапом необхідно доповнити алгоритм, додавши використання додаткових кнопок та світлодіодної індикації.

Висновки

У роботі досліджено методи управління пневматичними приводами на основі контролера Arduino, що дозволяє реалізувати гнучкі та економічні рішення для автоматизації пневматичних систем. Запропоновані підходи до моделювання включають математичне описання динаміки пневматичних компонентів, таких як циліндри і клапани, з урахуванням компресії повітря та нелінійних властивостей системи.

Інтеграція моделі з контролером Arduino забезпечує можливість реалізації алгоритмів регулювання у реальному часі, що дозволяє коригувати параметри роботи приводів залежно від вхідних сигналів і зворотного зв'язку. Використання візуалізації даних через серійний монітор або графічні інтерфейси підвищує точність діагностики і налаштування системи, сприяє швидкому виявленню аномалій та оптимізації роботи.

Отримані результати підтверджують ефективність запропонованих моделей для побудови систем управління з високою адаптивністю і стабільністю. Подальший розвиток досліджень передбачає вдосконалення алгоритмів управління з використанням методів штучного інтелекту та більш складних моделей динаміки пневматичних приводів для забезпечення підвищеної точності та швидкодії.

Список літератури

1. Широкий, Ю. В. Автоматизування об'єктів промислового виробництва пневматичними системами: метод. рекомендації до виконання розрахунково-графічної роботи. / Ю. В. Широкий, О.В. Торосян, Д. Р. Степаненко / Харків: Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2025. – 57 с.
2. Цвіркун, Л. І. Робототехніка та мехатроніка: навч. посіб. / Л. І. Цвіркун, Г. Гулер / Дніпро: Нац. гірн. ун-т, 2017. – 224с.
3. Губарев, О.П. Програмовані контролери в системах керування гідропневмоавтоматики: методичні вказівки до лабораторних робіт / О. П. Губарев, О. В. Левченко / Київ: НТУУ «КПІ», 2006. – 52 с.
4. TP201 Electropneumatics Basic level Workbook / Markus Pany, Sabine Scharf, Ralph-Christoph Weber, Frank Ebel - Festo Didactic SE, Reichbergst. 3,73770 Denkendorf, Germany, 2016. – 121 с.
5. Zhen, Z. Z. Structure and Application of Soft Pneumatic Actuators. / Z. Z. Zhen // Proceedings of the 2nd International Conference on Machine Learning and Automation. Springer, China. – 2024. – P. 191-196. <https://doi.org/10.54254/2755-2721/2024.18286>.
6. Симулятор роботи електричних схем Wokwi [Електронний ресурс] // Режим доступу: <https://wokwi.com/>
7. Середовище розробки Arduino IDE [Електронний ресурс] // Режим доступу: <https://docs.arduino.cc/software/ide-v2/tutorials/getting-started/ide-v2-downloading-and-installing/>
8. Середовище розробки пневматичних схем FluidSIM FESTO [Електронний ресурс] // Режим доступу: <https://www.art->

[systems.de/www/site/en/downloads/fluidsim6.html](https://www.art-systems.de/www/site/en/downloads/fluidsim6.html)

References

1. Shyrokyi, Yu. V. Automation of industrial production facilities with pneumatic systems: methodological recommendations for the implementation of calculation and graphic work / Yu. V. Shyrikyi, O. V. Torosian, D. R. Stepanenko / Kharkiv : National aerospace un-t M. E. Zhukovsky "Kharkiv. aviation. in-t.", 2025. – 57 p.
2. Tsvirkun, L.I. Robotics and mechatronics: textbook / L. I. Tsvirkun, G. Guler / Dnipro: NGU, 2017. – 224 p.
3. Gubarev, O.P. Programmable controllers in hydro-pneumatic automation control system: methodical instructions for laboratory work / O. P. Gubarev , O. V. Levchenko / Kyiv: NTUU "KPI", 2006. – 52 p.
4. TP201 Electropneumatics Basic level Workbook / Markus Pany, Sabine Scharf, Ralph-Christoph Weber, Frank Ebel - Festo Didactic SE, Reichbergst. 3,73770 Denkendorf, Germany, 2016. – 121 p.
5. Zhen, Z. Z. Structure and Application of Soft Pneumatic Actuators. / Z. Z. Zhen / Proceedings of the 2nd International Conference on Machine Learning and Automation. Springer, China, 2024. – pp. 191-196. <https://doi.org/10.54254/2755-2721/2024.18286>.
6. Wokwi electrical circuit simulator [Electronic resource] // Access mode: <https://wokwi.com/>
7. Arduino IDE pneumatic circuit development environment [Electronic resource] // Access mode: <https://docs.arduino.cc/software/ide-v2/tutorials/getting-started/ide-v2-downloading-and-installing/>
8. FluidSIM FESTO development environment [Electronic resource] // Access mode: <https://www.art-systems.de/www/site/en/downloads/fluidsim6.html>

Надійшла до редакції 30.10.2025, розглянута на редколегії 18.11.2025

Controlling pneumatic actuators via an Arduino controller: modeling and visualization approaches

This paper considers modern approaches to controlling pneumatic actuators using the Arduino microcontroller. Pneumatic actuators are widely used in automated systems, in particular in production lines, mechatronics and robotics, due to their simplicity of design, speed of response and safety. However, their effective control requires clear coordination of the electronic and pneumatic parts of the system, which involves modeling the dynamics of the actuator, building control algorithms and convenient visualization of the process. Arduino as an open hardware and software platform is an ideal tool for implementing prototypes of control systems. The study analyzes the main types of pneumatic actuators (single- and double-acting), electro-pneumatic valves and position sensors. The methodology for building a model of a pneumatic system in the Wokwi environment and its hardware implementation using Arduino is presented. Particular attention is paid to the development of control algorithms based on PI controllers, feedback logic and time delays that take into account the inertia and compressibility of air. The paper also describes methods for visualizing the states of a pneumatic actuator using the FluidSIM demo interface, as

well as simulation in the Wokwi environment, which allows creating interactive monitoring and debugging systems. Examples of integrating pressure sensors, position sensors and displays for reading parameters in real time are given. The purpose of the research is to increase the efficiency, reliability and convenience of controlling pneumatic systems based on inexpensive and open platforms. The results can be useful for engineers, students and developers of automated systems, and will also be used in educational laboratories to demonstrate the principles of operation of pneumatic mechanisms and digital control.

Keywords: pneumatic actuators, Arduino, modeling, visualization, automation, actuator control, microcontroller, Simulink, sensors.

Відомості про авторів:

Степаненко Денис Романович – старший викладач, кафедра теоретичної механіки, машинознавства та роботомеханічних систем, Національний аерокосмічний університет «Харківський авіаційний інститут», м. Харків, Україна. d.stepanenko@khai.edu, ORCID: <https://orcid.org/0000-0002-6765-096X>.

Торосян Олена Василівна – старший викладач, кафедра теоретичної механіки, машинознавства та роботомеханічних систем, Національний аерокосмічний університет «Харківський авіаційний інститут», м. Харків, Україна. o.tarasyan@khai.edu, ORCID: <https://orcid.org/0000-0002-7389-6093>.

About the Authors:

Stepanenko Denys Romanovich – Senior Lecturer, Department of Theoretical Mechanics, Engineering and Robomechanical Systems, National Aerospace University “Kharkiv Aviation Institute”, Kharkiv, Ukraine. e-mail: d.stepanenko@khai.edu, ORCID: <https://orcid.org/0000-0002-6765-096X>.

Torosian Olena Vasilivna – Senior Lecturer, Department of Theoretical Mechanics, Engineering and Robomechanical Systems, National Aerospace University “Kharkiv Aviation Institute”, Kharkiv, Ukraine; e-mail: o.tarasyan@khai.edu, ORCID: <https://orcid.org/0000-0002-7389-6093>.