

ДОДАТОК А
Апробація результатів роботи

ЗБІРНИК НАУКОВИХ ПРАЦЬ

З МАТЕРІАЛАМИ V МІЖНАРОДНОЇ НАУКОВОЇ КОНФЕРЕНЦІЇ

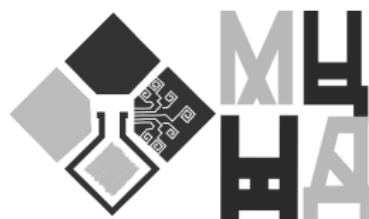
6 ЧЕРВНЯ 2025 РІК

М. КРОПИВНИЦЬКИЙ, УКРАЇНА

**«ПЕРІОД ТРАНСФОРМАЦІЙНИХ ПРОЦЕСІВ
В СВІТОВІЙ НАУЦІ: ЗАДАЧІ ТА ВИКЛИКИ»**



ЗБІРНИК НАУКОВИХ
ПРАЦЬ З МАТЕРІАЛАМИ
V МІЖНАРОДНОЇ
НАУКОВОЇ КОНФЕРЕНЦІЇ



ПЕРІОД ТРАНСФОРМАЦІЙНИХ ПРОЦЕСІВ В СВІТОВІЙ НАУЦІ: ЗАДАЧІ ТА ВИКЛИКИ

| 6 червня 2025 рік
м. Кропивницький, Україна

Вінниця, Україна
«UKRLOGOS Group»
2025

УДК 082:001
П 26



Організація, від імені якої випущено видання:

ГО «Міжнародний центр наукових досліджень»

Номер запису організації в Єдиному реєстрі громадських об'єднань: 1499141.

Голова оргкомітету: Сотник С.Г.

Верстка: Білоус Т.В.

Дизайн: Бондаренко І.В.

Рекомендовано до видання Вченою Радою Інституту науково-технічної інтеграції та співпраці. Протокол № 22 від 05.06.2025 року.



Конференцію зареєстровано Державною науковою установою у сфері управління Міністерства освіти і науки «Український інститут науково-технічної експертизи та інформації» в базі даних науково-технічних заходів України на поточний рік та бюлетені «План проведення наукових, науково-технічних заходів в Україні» (Посвідчення № 103 від 06.01.2025).

Збірник наукових праць з матеріалами конференції видано офіційно суб'єктом видавничої справи зі **Свідоцтвом ДК № 7860 від 22.06.2023**.

Матеріали конференції знаходяться у відкритому доступі на умовах ліцензії Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0).

П 26 **Період трансформаційних процесів в світовій науці: задачі та виклики:** збірник наукових праць з матеріалами V Міжнародної наукової конференції, м. Кропивницький, 6 червня, 2025 р. / Міжнародний центр наукових досліджень. — Вінниця: ТОВ «УКРЛОГОС Груп, 2025. — 478 с.

ISBN 978-617-8312-60-2

DOI 10.62731/mcnd-06.06.2025

Викладено матеріали учасників V Міжнародної наукової конференції «Період трансформаційних процесів в світовій науці: задачі та виклики», яка відбулася 6 червня 2025 року у місті Кропивницький.

УДК 082:001

© Колектив учасників конференції, 2025

© ГО «Міжнародний центр наукових досліджень», 2025

ISBN 978-617-8312-60-2

© ТОВ «УКРЛОГОС Груп», 2025

СЕКЦІЯ XII. АТОМАТИЗАЦІЯ ТА ПРИЛАДОБУДУВАННЯ

DEVELOPMENT OF A STRUCTURAL SCHEME FOR AUTOMATIC DOSING OF LIQUID COMPONENTS

Shrubkovskiy Yevhen Viktorovich

higher education seeker of the Faculty of ACT
Kharkiv National University of Radio Electronics, Ukraine

Sotnik Svitlana Viktorivna

ORCID ID: 0000-0002-6035-2388

PhD, Associate Professor of Department of Computer-Integrated
Technologies, Automation and Robotics
Kharkiv National University of Radio Electronics, Ukraine

Automation of production processes is one of the most important trends in modern industry, allowing companies to remain competitive in the face of globalization and increasing demands for product quality [1, 2]. In modern production, the accuracy of dosing liquid components plays a critical role in ensuring product quality [3]. Inaccuracies in dosing can lead to defects, economic losses, and non-compliance with quality standards. Automation of the dosing process allows for high accuracy and repeatability of results, which cannot be ensured through manual dosing.

The development of a structural diagram for the automatic dosing of liquid components (ADLC) is an extremely relevant topic in modern industry and scientific-technical progress for the following reasons: ensuring product quality, economic efficiency of automatic dosing systems, minimizing personnel contact with potentially hazardous substances, etc.

The key goal is to develop an effective automated system capable of providing accurate dosing of liquid components according to technological requirements. One of the significant stages of this process is the construction of a structural diagram (Fig. 1), which defines the composition, relationships, and operating principles of the main elements of the system.

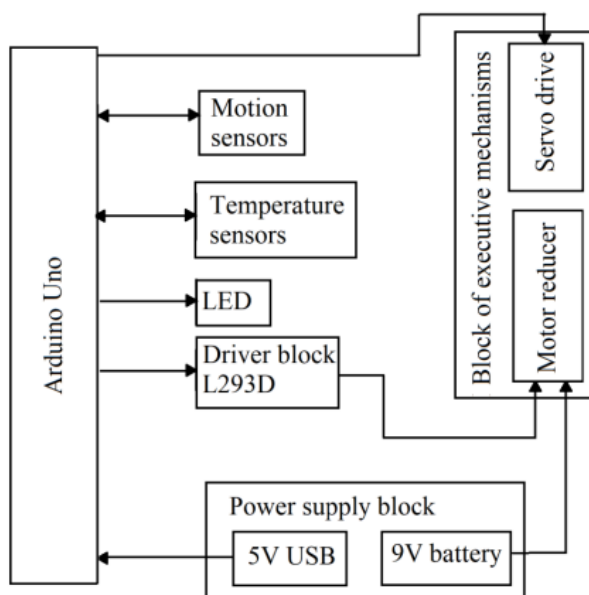


Fig. 1. General structural scheme of the subsystem for automatic dosing of liquid components

The presented diagram (Fig. 1) demonstrates the architecture of the automatic control system, built on the Arduino Uno microcontroller as the central control element, the main components of the system are:

1. The central controller – Arduino Uno serves as the main computing unit of the system, coordinating the operation of all connected components through digital and analog inputs/outputs.

2. Sensor subsystem:

– motion sensors – intended for detecting the operation of the conveyor, they transmit signals about the movement status to Arduino via bidirectional communication lines;

– temperature sensors – monitor the temperature of the liquid component. Such sensors are critically important for quality control of the dosing of liquid components [4].

To detect the movement of the conveyor in the automatic dosing system, it is advisable to choose an infrared optical sensor of the TCRT5000 type. For the study, the TMP36 temperature sensor was chosen due to its ease of use, high accuracy, and stability of operation over a wide temperature range. It outputs an analog signal that is linearly dependent on temperature, making

data processing with a microcontroller simpler. The TMP36 does not require calibration and has low power consumption.

3. The indication system – LED elements provide visual indication of the system's operating status, signaling the operating modes, errors, or completion of dosing cycles.

4. The executive subsystem is the L293D driver block, which is a key element for controlling electric motors and provides:

- amplification of control signals from Arduino;
- bidirectional control of motor rotation;
- galvanic isolation between logic and power circuits.

The selection of the L293D driver for the ADLC system is due to its technical characteristics and functional capabilities, which fully meet the requirements of similar automated solutions. The L293D allows for controlling the direction and speed of DC motors, which is necessary for precise liquid dosing through a valve. With the ability to independently control two channels, it provides flexibility in implementing various operating modes of the system. An important factor is also the compatibility of the L293D with microcontrollers, which significantly simplifies integration into the overall architecture of the dosing control system.

The JGB37-520 motor-reducer was chosen because it provides the widest range of rotations (from 10 to 1000 RPM), allowing for flexible adjustment of the liquid feed speed according to dosing needs. This motor also operates over a wide voltage range (6–24 V), which enhances its versatility in various power supply circuits.

5. Block of executive mechanisms:

- servo drive, which ensures precise positioning of system elements (dosing devices);
- motor-reducer, which creates torque for the drive of the main dosing mechanisms.

The SG90 servo drive was chosen for its compactness, low energy consumption, and sufficient positioning accuracy, which allows for effective control of mechanisms for opening or adjusting the flow of liquid in an automatic dosing system. Its compatibility with microcontrollers and ease of control make it convenient for integration into automated solutions.

6. The power system includes dual-channel power supply:

- 5V USB – for powering logic components (Arduino, sensors, LEDs);

2. Sotnik S. V. (2024) Development of automated control system and registration of metal in continuous casting. *Radio Electronics, Computer Science, Control*, (3), 197–211. Removed from: https://www.researchgate.net/publication/385539711_DEVELOPMENT_OF_AUTOMATED_CONTROL_SYSTEM_AND_REGISTRATION_OF_METAL_IN_CONTINUOUS_CASTING
3. Ajiga D. & et al. (2024) The role of software automation in improving industrial operations and efficiency. *International Journal of Engineering Research Updates*, 7(1), 22–35. Removed from: <https://www.researchgate.net/profile/Daniel-Ajiga/publication/383410390>
4. Kolbasa O. R. & et al. (2025) The significance and necessity of automating the selection of sensors and actuators. «*Computer-integrated technologies, automation and robotics*» CITAR-2025, 63–67. Removed from: <https://openarchive.nure.ua/entities/publication/67280cb0-4071-4fe5-b653-0bcf663fb5fe/full>

ДОДАТОК Б

Програмний код

```
//Розпіновка
const int tempPin = A0; //Датчик температури
const int irSensorPin = 3; //Інфрачервоний датчик перешкод
const int ledPin = 4; //Світлодіод
const int motorPin1 = 5; //Мотор-редуктор (+)
const int motorPin2 = 6; //Мотор-редуктор (-)
const int pumpPin1 = 7; //Насос (+)
const int pumpPin2 = 8; //Насос (-)
//Оголошення змінних
bool ledState = HIGH; //Стан світлодіода
bool systemRunning = true; //Стан системи
unsigned long lastTempCheck = 0; //Час останньої перевірки температури
unsigned long irDetectedTime = 0; //Час виявлення перешкоди
bool irHandled = false; //Чи оброблено виявлення перешкоди
bool pumpRunning = false; //Чи працює насос
unsigned long pumpStartTime = 0; //Час запуску насосу

void setup() {
    Serial.begin(9600); //Ініціалізація послідовного порту
    pinMode(ledPin, OUTPUT);
    pinMode(irSensorPin, INPUT); //Інфрачервоний датчик
    pinMode(motorPin1, OUTPUT);
    pinMode(motorPin2, OUTPUT);
    pinMode(pumpPin1, OUTPUT);
    pinMode(pumpPin2, OUTPUT);
    digitalWrite(ledPin, HIGH); //Світлодіод увімкнено
    digitalWrite(motorPin1, HIGH); //Мотор-редуктор увімкнено
    digitalWrite(motorPin2, LOW);
    digitalWrite(pumpPin1, LOW); //Насос вимкнено
    digitalWrite(pumpPin2, LOW);
}

void loop() {
    //Перевірка температури кожні 500 мс
    if (millis() - lastTempCheck >= 500) {
        float temperature = readTemperature();
        lastTempCheck = millis();
        //Вивід значень температури до послідовного порту
    }
}
```

```

Serial.print("Temperature: ");
Serial.print(temperature);
Serial.println("°C");
//Перевірка значення температури
if (temperature < 0 || temperature > 40) {
    systemRunning = false; //Зупинка системи
    stopAll(); //Зупинка моторів
    blinkLED(); //Блимання світлодіода
    return; //Вихід з циклу
} else {
    systemRunning = true; //Система працює
    digitalWrite(ledPin, HIGH); //Світлодіод увімкнено
}
}

//Робота системи
if (systemRunning) {
    //Перевірка інфрачервоного датчика
    if (digitalRead(irSensorPin) == LOW && !irHandled) { //Перешкоду виявлено
        if (irDetectedTime == 0) {
            irDetectedTime = millis(); //Початок відліку часу
            digitalWrite(motorPin1, LOW); //Зупинка мотор-редуктору
            digitalWrite(motorPin2, LOW);
            digitalWrite(pumpPin1, HIGH); //Увімкнення насосу
            digitalWrite(pumpPin2, LOW);
            pumpRunning = true;
            pumpStartTime = millis();
        }
    }

    //Перевка часу роботи насосу
    if (pumpRunning && millis() - pumpStartTime >= 2000) {
        digitalWrite(pumpPin1, LOW); //Вимикаємо насос
        digitalWrite(pumpPin2, LOW);
        pumpRunning = false;
        irHandled = true; //Виявлення оброблено
        //Відновлення роботи мотор-редуктору одразу після зупинки насоса
        digitalWrite(motorPin1, HIGH);
        digitalWrite(motorPin2, LOW);
    }

    //Якщо перешкода все ще виявлена після 2 секунд роботи насосу + 1 секунду
    if (irHandled && digitalRead(irSensorPin) == LOW) {
        if (millis() - irDetectedTime >= 3000) { //3 секунди від виявлення

```

```

        systemRunning = false; //Зупинка системи
        stopAll(); //Зупинка моторів
        blinkLED(); //Блимання світлодіоду
    }
} else if (irHandled && digitalRead(irSensorPin) == HIGH) {
    //Скидання стану, якщо перешкода зникла
    irHandled = false;
    irDetectedTime = 0;
}
}
}
//Зчитування температури
float readTemperature() {
    int sensorValue = analogRead(tempPin);
    float voltage = sensorValue * (5.0 / 1023.0); //Конвертація в вольти
    float temperature = (voltage - 0.5) * 100; //Конвертація в градуси Цельсія
    return temperature;
}
//Зупинка всього
void stopAll() {
    digitalWrite(motorPin1, LOW); //Зупинка мотор-редуктора
    digitalWrite(motorPin2, LOW);
    digitalWrite(pumpPin1, LOW); //Зупинка насоса
    digitalWrite(pumpPin2, LOW);
}
//Блимання світлодіоду
void blinkLED() {
    unsigned long currentTime = millis();
    if (currentTime % 1000 < 500) {
        digitalWrite(ledPin, HIGH);
    } else {
        digitalWrite(ledPin, LOW);
    }
}
}

```

ДОДАТОК В
Демонстраційний матеріал

[illegible]