

ДОДАТОК А
Результати роботи у статті

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
(КІТАР)



МАТЕРІАЛИ

II Всеукраїнської конференції
«Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки»
(Computer-integrated technologies, automation and robotics)

CITAR'25

16-17 травня 2025

[електронне видання]

Харків 2025

УДК: 005:004.896:62-65:338.3

Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки 2025: матеріали II-ої Всеукраїнської конференції, Харків, 16-17 травня 2025.: тези доповідей / [редкол. І.Ш. Невлюдов (відповідальний редактор)].-Харків: [електронний друк], 2025. – 132 с.

У збірник включені тези доповідей, які присвячені сучасним автоматизованим технологіям Industry 4.0 та їх впровадження; інформаційні управляючі системи технологічного призначення; математичні методи в системах автоматизації; розробка та програмування в робототехніці; штучний інтелект та машинне навчання в автоматизації; інтеграція технологій у виробництві та промисловості; сенсорні технології та взаємодія людини з роботами в Industry 5.0; ефективність використання роботизованих систем у виробництві; етика та правові аспекти в робототехніці; Інтернет речей та Інтегровані системи в комп'ютерно-інтегрованих технологіях, автоматизації та робототехніки; технологічні виклики та інновації у світі робототехніки.

Редакційна колегія: І.Ш. Невлюдов, В.В. Євсєєв.

Computer-integrated technologies, automation and robotics 2025: Proceedings of II st All-Ukrainian Conference, Kharkiv, May 16-17, 2025: Theses of Reports / [Ed. I.Sh. Nevlyudov (chief editor).] .- Kharkiv .: [electronic version], 2025. - 132 p.

The collection includes abstracts devoted to modern automated technologies of Industry 4.0 and their implementation; information control systems for technological purposes; mathematical methods in automation systems; development and programming in robotics; artificial intelligence and machine learning in automation; integration of technologies in production and industry; sensor technologies and human interaction with robots in Industry 5.0; efficiency of using robotic systems in production; ethics and legal aspects in robotics; Internet of Things.

Editorial board: Igor.Sh. Nevlyudov, Vladyslav.V. Yevsieiev

КОМІТЕТ КОНФЕРЕНЦІЇ

Голова комітету конференції	Невлюдов Ігор Шакирович , доктор технічних наук, професор, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Секретар конференції	Євсєєв Владислав В'ячеславович , доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Технічний секретар	Самойленко Ганна Юріївна , асистент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Редакційна колегія:	<p>Филипенко Олександр Іванович, доктор технічних наук, професор, декан факультету Автоматики та комп'ютеризованих технологій, Харківського національного університету радіоелектроніки.</p> <p>Цимбал Олександр Михайлович, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.</p> <p>Ромашов Юрій Володимирович доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.</p> <p>Косенко Віктор Васильович, доктор технічних наук, професор, зам. директора Державного підприємство «Південний державний проектно- конструкторський та науково-дослідний інститут авіаційної промисловості».</p> <p>Замірець Микола Васильович, доктор технічних наук, професор, директор Державного підприємства Науково-дослідного технологічного інституту приладобудування.</p> <p>Свищ Володимир Митрофанович, доктор технічних наук, професор, радник директора Державне науково-виробниче підприємство «Об'єднання Комунар».</p> <p>Кухаренко Дмитро Володимирович, кандидат технічних наук, доцент кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського</p> <p>Фурманова Наталія Іванівна, кандидат технічних наук, доцент, декан факультета Інформаційної безпеки та електронних комунікацій, Національного університету «Запорізька політехніка».</p>

РОЛЬ SCADA-СИСТЕМИ З ВИКОРИСТАННЯМ КОНЦЕПЦІЇ ІОТ

Юрченко О.Д.

Харківський національний університет радіоелектроніки

Україна, 61166, Харків, пр. Науки 14

E-mail: oleh.iurchenko@nure.ua

Анотація: У даній роботі буде розглянуто роль SCADA-систем у контексті концепції Індустрії 4.0, яка базується на впровадженні сучасних цифрових технологій, зокрема з інтеграцією Інтернету речей (ІоТ), що підкреслює значення SCADA-систем для сучасного виробництва.

Ключові слова: SCADA-система, Індустрія 4.0, Інтернет речей, автоматизація, моніторинг, управління, виробничий процес.

THE ROLE OF THE SCADA SYSTEM USING THE IOT CONCEPT

Yurchenko Oleg Dmytrovych

Kharkiv National University of Radio Electronics

Ukraine, 61166, Kharkiv, Nauky av., 14

E-mail: oleh.iurchenko@nure.ua

Annotation: This article considers the role of SCADA systems in the context of the Industry 4.0 concept, which is based on the introduction of modern digital technologies, in particular with the integration of the Internet of Things (IoT), which emphasizes the importance of SCADA systems for modern production.

Key words: SCADA system, Industry 4.0, Internet of Things, automation, monitoring, control, production process.

АКТУАЛЬНІСТЬ РОБОТИ. Сучасні виробничі лінії швидко еволюціонують під впливом концепції Індустрії 4.0, яка базується на впровадженні цифрових технологій, таких як Інтернет речей (ІоТ), хмарні обчислення, штучний інтелект та великі дані. У цьому контексті SCADA-системи, які є ключовим аспектом автоматизованого управління виробничими процесами, також зазнають змін. Розширення та вдосконалення функціональних можливостей SCADA-систем за рахунок інтеграції ІоТ пристроїв створює нові можливості для більш гнучкого, ефективного та безпечного керування.

ВСТУП. Сучасна промисловість швидко розвивається, впроваджуючи новітні технології, які формують концепцію Індустрії 4.0. Одним із ключових елементів цієї концепції є SCADA-системи (Supervisory Control and Data Acquisition), які забезпечують моніторинг, управління та оптимізацію виробничих процесів. Інтеграція SCADA-систем із технологіями Інтернету речей (ІоТ) відкриває нові можливості для автоматизації, підвищення ефективності та гнучкості виробництва.

SCADA-системи традиційно використовуються для збору даних і управління технологічними процесами в режимі реального часу. Завдяки впровадженню ІоТ, такі системи стають більш адаптивними, дозволяючи інтегрувати величезну кількість датчиків та пристроїв, а також забезпечувати обробку даних у хмарі. Це сприяє покращенню ефективності виробничих ліній, зниженню витрат та підвищенню безпеки [1].

У статті буде розглянуто теоретичні аспекти SCADA-систем, їх компоненти, типи та особливості інтеграції з ІоТ. Також буде представлено порівняння різних типів SCADA-систем та описано переваги використання Node-Red як базової платформи для їх створення.

1 ОГЛЯД SCADA-систем. SCADA-системи - це високорозподілені системи, які використовуються для управління географічно розподіленими активами, часто розкиданими на тисячі квадратних кілометрів, де централізований збір даних і контроль є критично важливими для роботи системи. Вони використовуються в розподільчих системах, таких як системи водопостачання та збору стічних вод, нафто- і газопроводи, електромережі та залізничні транспортні системи. Центр управління SCADA виконує централізований моніторинг і управління польовими об'єктами через міжміські мережі зв'язку, включаючи моніторинг аварійних сигналів і обробку даних про стан. На основі інформації, отриманої з віддалених станцій, автоматизовані або керовані оператором диспетчерські команди можуть передаватися на віддалені пристрої управління станціями, які часто називають польовими пристроями. Польові пристрої керують локальними операціями, такими як відкриття і закриття клапанів і вимикачів, збір даних з сенсорних систем і моніторинг локального середовища на наявність тривожних станів [1].

Одним із ключових компонентів будь-якої SCADA-системи є набір датчиків, які збирають інформацію з об'єкта, а також спеціалізовані багатопроцесорні обчислювальні комплекси з високою відмовостійкістю, що забезпечують обробку даних та управління в умовах суворого реального часу. Сучасні SCADA-системи беруть свій початок від систем телеметрії та сигналізації, які були їхніми прототипами на початкових етапах розвитку автоматизованих систем управління [2].

Усі сучасні SCADA-системи складаються з трьох основних структурних компонентів, а саме віддаленого терміналу, диспетчерського пункту керування та комунікаційної системи.

Віддалений термінальний пристрій (Remote Terminal Unit, RTU) виконує завдання управління в реальному часі.

Головний термінал (Master Terminal Unit, MTU) або диспетчерський пункт (Master Station, MS) виконує обробку даних і управління на високому рівні, зазвичай у режимі м'якого реального часу [2].

Комунікаційна система (Communication System, CS) забезпечує передачу даних між віддаленими об'єктами чи терміналами та центральним інтерфейсом оператора.

На рисунку 1 наведена спрощена структурна схема типової SCADA-системи.

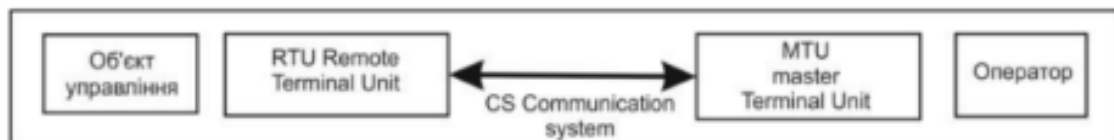


Рисунок 1 – Структурна схема SCADA-системи [2]

Основні елементи SCADA-систем:

- програмовані логічні контролери (PLC);
- віддалений термінальний пристрій;
- людино-машинний інтерфейс;
- сервери;
- бази даних;

Програмовані логічні контролери відповідають за безпосереднє управління технологічними процесами, збір даних від датчиків та передачу команд виконавчим механізмам. RTU забезпечують збір даних з віддалених об'єктів та передачу їх до центральної системи. НМІ надає операторам зручний графічний інтерфейс для моніторингу та управління процесами. Сервери виконують обробку, зберігання та управління даними, забезпечуючи взаємодію між

компонентами системи. Бази даних зберігають історичні та поточні дані про процеси, що дозволяє аналізувати та оптимізувати роботу системи.

Основними функціями SCADA-систем виділяють наступні:

- збір даних, отримання інформації від обладнання технологічного процесу та управління ним на відстані, а також ведення бази даних у реальному часі [2];
- створення зручного для оператора графічного інтерфейсу для моніторингу та керування процесами, а також представлення інформації з бази даних у зручному для аналізу вигляді;
- автоматизація робочих процесів для полегшення прийняття рішень оператором.
- виконання розрахунків ефективності виробництва, аналізу статистики процесу та продуктивності обладнання;
- виконання завдань, таких як блокування, некритичне регулювання та інші функції управління;
- генерація повідомлень і сигналів тривоги автоматично у випадку збоїв чи аномалій у системі;
- підготовка звітів, оперативних зведень, рапортів та іншої експлуатаційної документації;
- зберігання історії даних, тривог та дій оператора для подальшого аналізу;
- розмежування прав користувачів за категоріями, забезпечення безпеки та контроль дій оператора;
- забезпечення резервування критичних елементів системи, таких як сервери, мережі та клієнтські інтерфейси;
- горизонтальний обмін даними з іншими системами автоматизованого управління та передача інформації на вищі рівні управління.

Ці функції забезпечують надійність, гнучкість і ефективність роботи сучасних SCADA-систем, що робить їх незамінними для управління технологічними процесами.

Сучасні SCADA-системи мають відповідати низці вимог, які спрямовані на забезпечення їхньої надійності, функціональності та зручності використання. Основними аспектами, що турбують розробників SCADA-систем, є такі:

- забезпечення стабільного функціонування в умовах промислового середовища, що включає високі вимоги до стійкості системи та захисту від кіберзагроз;
- наявність великої кількості драйверів для роботи з програмованими логічними контролерами (PLC), віддаленими терміналами (RTU) і розподіленими системами вводу-виводу [3];
- наявність вбудованих інструментів для обробки та візуалізації даних, широкі можливості створення графічного та керуючого інтерфейсу;
- зручність у розширенні системи за рахунок різних типів вузлів (сервери, клієнти, runtime, development, web-клієнти, тонкі клієнти, підтримка мобільних платформ iOS/Android) та гнучкої системи ліцензування [3];
- інтуїтивно зрозумілий інтерфейс, якісна технічна документація та надійна технічна підтримка;
- регулярне оновлення системи для забезпечення сумісності з новими версіями операційних систем та актуальними нововведеннями для кібербезпеки.

2 РОЛЬ SCADA-СИСТЕМ В ІНДУСТРІЇ 4.0. На сьогоднішній день є необхідним перехід від традиційних SCADA-систем у кіберфізичні системи (CPS), які забезпечують інтеграцію з сучасними технологіями автоматизації, розширення можливостей збору, аналізу та управління даними.

Кіберфізичні системи забезпечують взаємодію між фізичними пристроями (датчиками, виконавчими механізмами) та комп'ютерними алгоритмами, які працюють у зворотному зв'язку.

Роль SCADA у CPS полягає у інтеграції великої кількості даних з польових пристроїв, наданні їх в уніфікованому вигляді для аналізу, дозволі автоматизувати прийняття рішень та адаптуванні виробництва в режимі реального часу.

Люди залишаються ключовими елементами у прийнятті стратегічних рішень, особливо в плануванні та управлінні виробничими циклами. Використання автоматизації не замінює людину, а розширює її можливості, забезпечуючи гнучкість у прийнятті рішень і контроль за складними процесами.

Архітектура SCADA у контексті Індустрії 4.0 описана по рівням [4]:

- польовий рівень (збір та передача інформації х пристроїв);
- контролери (прийм даних, локальне управління, передача на SCADA);
- SCADA (збір та аналіз даних, автоматизація процесів, попередження про аварійні ситуації);
- бізнес-рівень (система управління ресурсами виробництва ERP).

На рисунку 2 наведено структуру збору даних і оперативного диспетчерського управління SCADA-системи.



Рисунок 2 – Структуру збору даних і оперативного диспетчерського управління SCADA-системи

ВИСНОВКИ. У даній статті було розглянуто актуальність використання SCADA-систем у контексті Індустрії 4.0. Зокрема, було проаналізовано роль SCADA-систем як основного інструменту для моніторингу, управління та оптимізації технологічних процесів у сучасних виробничих лініях.

Також було проаналізовано виклики, з якими стикаються розробники та користувачі SCADA-систем, такі як висока вартість впровадження, складність інтеграції з існуючим обладнанням та забезпечення кібербезпеки.

Таким чином, у статті доведено, що SCADA-системи є незамінним компонентом сучасного виробництва. SCADA-системи забезпечують автоматизацію, адаптивність і ефективність, що дозволяє їм відповідати вимогам Індустрії 4.0 і сприяти сталому розвитку підприємств у різних галузях промисловості.

ЛІТЕРАТУРА

1. Guide to Industrial Control Systems (ICS) Security [Електронний ресурс] / – Режим доступу: <https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-82r2.pdf> (дата звернення: 18.12.2024). – Назва з екрана.
2. Автоматизована система керування технологічними процесами в SCADA системі TRACE MODE 6 : навч. посіб. / І. Ш. Невлюдов, А. О. Андрусевич, В. В. Євсєєв, та ін. – Кривий Ріг : Криворізький коледж НАУ, 2018 р. – 320 с.
3. SCADA система [Електронний ресурс] / Семенюк Ганна. Режим доступу: <https://digitap.com.ua/scada-systema-shho-cze-take/> (дата звернення: 18.12.2024).
4. Research on the SCADA system constructing methodology based on SOA [Електронний ресурс] / L. Wang – Режим доступу: https://www.researchgate.net/publication/290759633_Research_on_the_SCADA_system_constructing_methodology_based_on_SOA (дата звернення: 18.12.2024).
5. Архітектурні рішення та інтеграція [Електронний ресурс] / – Режим доступу: <http://edu.asu.in.ua/mod/book/tool/print/index.php?id=107> (дата звернення: 18.12.2024).
6. Supervisory Control and Data Acquisition (SCADA) Systems for Industrial Automation and Control Systems in Industry 4.0 [Електронний ресурс] / Florin Vasile Popescu – Режим доступу: https://www.researchgate.net/publication/367066607_Supervisory_Control_and_Data_Acquisition_SCADA_Systems_for_Industrial_Automation_and_Control_Systems_in_Industry_40 (дата звернення: 18.12.2024).

ДОДАТОК Б

Програмна реалізація для ESP32

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <WebServer.h>
#include <DHT.h>
#include <ESP32Servo.h>
#include <ArduinoJson.h>
const char* ssid = "MEGA_DESTROYER_2_4G";
const char* password = "gribova01";
WebServer server(80);
String nodeRedUrl = "http://192.168.1.102:1880/telemetry";
#define DHTPIN 18
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
#define TRIG_PIN 14
#define ECHO_PIN 27
const int SERVO_PIN1 = 17;
const int SERVO_PIN2 = 5;
const int SERVO_PIN3 = 19;
Servo s1, s2, s3;
int a1 = 90, a2 = 90, a3 = 90;
unsigned long lastSend = 0;
const unsigned long INTERVAL = 5000; // ms
void connectWiFi() {
    Serial.print("Connecting to WiFi...");
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
```

```
    delay(500);
    Serial.print(".");
}
Serial.println("\nWiFi connected! IP: " + WiFi.localIP().toString());
}
float getDistanceCM() {
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    long duration = pulseIn(ECHO_PIN, HIGH, 30000);
    float distance = duration * 0.034 / 2;
    if (duration == 0) return -1;
    return distance;
}
void setup() {
    Serial.begin(115200);
    delay(300);
    dht.begin();
    s1.attach(SERVO_PIN1);
    s2.attach(SERVO_PIN2);
    s3.attach(SERVO_PIN3);
    pinMode(TRIG_PIN, OUTPUT);
    pinMode(ECHO_PIN, INPUT);
    connectWiFi();
    server.on("/command", HTTP_POST, handleCommand);
    server.begin();
    Serial.println("HTTP server started!");
}
```

```

void sendTelemetry() {
  if (WiFi.status() != WL_CONNECTED) return;

  float hum = dht.readHumidity();
  float temp = dht.readTemperature();
  float distance = getDistanceCM();
  if (isnan(temp) || isnan(hum)) {
    Serial.println("DHT read error");
    return;
  }
  StaticJsonDocument<256> doc;
  doc["temp"] = temp;
  doc["hum"] = hum;
  doc["servo1"] = a1;
  doc["servo2"] = a2;
  doc["servo3"] = a3;
  doc["distance_cm"] = distance;
  doc["uptime_s"] = millis() / 1000;
  String json;
  serializeJson(doc, json);
  HTTPClient http;
  http.begin(nodeRedUrl);
  http.addHeader("Content-Type", "application/json");
  int code = http.POST(json);
  http.end();
  Serial.print("Sent telemetry -> HTTP code: ");
  Serial.println(code);
}

void handleCommand() {
  if (!server.hasArg("plain")) {

```

```

server.send(400, "text/plain", "No data");
return;
}

String body = server.arg("plain");
StaticJsonDocument<256> cmd;
DeserializationError err = deserializeJson(cmd, body);
if (err) {
  server.send(400, "text/plain", "JSON parse error");
  return;
}
if (cmd.containsKey("servo1")) {
  a1 = constrain(cmd["servo1"].as<int>(), 0, 180);
  s1.write(a1);
}
if (cmd.containsKey("servo2")) {
  a2 = constrain(cmd["servo2"].as<int>(), 0, 180);
  s2.write(a2);
}
if (cmd.containsKey("servo3")) {
  a3 = constrain(cmd["servo3"].as<int>(), 0, 180);
  s3.write(a3);
}
server.send(200, "application/json", "{\"status\":\"ok\"}");
}

void loop() {
  server.handleClient();
  if (millis() - lastSend > INTERVAL) {
    lastSend = millis();
    sendTelemetry();
  }
}

```

}
}

ДОДАТОК В
Програмна реалізація для Node-RED

```
[
  {
    "id": "0b83b3a7aa84f3c1",
    "type": "http in",
    "z": "fb9a3a66b864403d",
    "name": "",
    "url": "/telemetry",
    "method": "post",
    "upload": false,
    "skipBodyParsing": false,
    "swaggerDoc": "",
    "x": 100,
    "y": 420,
    "wires": [
      [
        "a08cbaa84caa83ef"
      ]
    ]
  },
  {
    "id": "9b6e10dff62d5614",
    "type": "debug",
    "z": "fb9a3a66b864403d",
    "name": "debug 2",
    "active": true,
    "tosidebar": true,
```

```
"console": false,  
"tostatus": false,  
"complete": "false",  
"statusVal": "",  
"statusType": "auto",  
"x": 460,  
"y": 420,  
"wires": []  
},  
{  
  "id": "b8e8dea08737731c",  
  "type": "http response",  
  "z": "fb9a3a66b864403d",  
  "name": "",  
  "statusCode": "",  
  "headers": {},  
  "x": 450,  
  "y": 480,  
  "wires": []  
},  
{  
  "id": "58ff538a9843bb58",  
  "type": "ui_slider",  
  "z": "fb9a3a66b864403d",  
  "name": "",  
  "label": "Servo1",  
  "tooltip": "",  
  "group": "129eb55ea17bdbce",  
  "order": 5,  
  "width": 0,
```

```
"height": 0,
"passthru": true,
"outs": "all",
"topic": "topic",
"topicType": "msg",
"min": 0,
"max": "180",
"step": 1,
"className": "",
"x": 100,
"y": 920,
"wires": [
  [
    "ab39abd290e52525"
  ]
],
},
{
  "id": "898ce854f961b4bc",
  "type": "http request",
  "z": "fb9a3a66b864403d",
  "name": "",
  "method": "use",
  "ret": "txt",
  "paytoqs": "ignore",
  "url": "http://192.168.1.104/command",
  "tls": "",
  "persist": false,
  "proxy": "",
  "insecureHTTPParser": false,
```

```
"authType": "",
"senderr": false,
"headers": [],
"x": 510,
"y": 920,
"wires": [
  [
    "932cee2c47aa8a1c"
  ]
],
},
{
  "id": "932cee2c47aa8a1c",
  "type": "debug",
  "z": "fb9a3a66b864403d",
  "name": "debug 3",
  "active": true,
  "tosidebar": true,
  "console": false,
  "tostatus": false,
  "complete": "false",
  "statusVal": "",
  "statusType": "auto",
  "x": 680,
  "y": 920,
  "wires": []
},
{
  "id": "5e5f3ac7d04f4c40",
  "type": "ui_gauge",
```

```
"z": "fb9a3a66b864403d",
"name": "",
"group": "e4f3410ae57e4461",
"order": 6,
"width": 0,
"height": 0,
"gtype": "gage",
"title": "Temperature",
"label": "*C",
"format": "{{msg.payload.temp}}",
"min": 0,
"max": "100",
"colors": [
  "#00b500",
  "#e6e600",
  "#ca3838"
],
"seg1": "",
"seg2": "",
"diff": false,
"className": "",
"x": 750,
"y": 300,
"wires": []
},
{
  "id": "a08cbaa84caa83ef",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "parser",
```

```

"func": "\nlet data = msg.payload;\nif (typeof data === 'string') {\ntry
{\ndata = JSON.parse(data);\n} catch(e) {\nnode.warn('JSON parse error: ' +
e);\nreturn null;\n}\n}\n\nmsg.payload = {\ntemp: Number(data.temp),\nhum:
Number(data.hum),\nservo1:          Number(data.servo1),\nservo2:
Number(data.servo2),\nservo3:          Number(data.servo3),\ndistance_cm:
Number(data.distance_cm),\nuptime_s:  Number(data.uptime_s)    ||    0,\nraw:
data\n};\nmsg.topic = 'telemetry';\nreturn msg;"}
"outputs": 1,
"timeout": 0,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 290,
"y": 420,
"wires": [
  [
    "9b6e10dff62d5614",
    "b8e8dea08737731c",
    "12119f3beb5b3328",
    "ebda825676e2e82a",
    "a6ed352648ede64e",
    "39165f96c1e6d2e2"
  ]
]
},
{
  "id": "12119f3beb5b3328",
  "type": "link out",
  "z": "fb9a3a66b864403d",

```

```
"name": "link out 2",
"mode": "link",
"links": [
  "c131dc420aba0d39"
],
"x": 405,
"y": 320,
"wires": []
},
{
  "id": "c131dc420aba0d39",
  "type": "link in",
  "z": "fb9a3a66b864403d",
  "name": "link in 2",
  "links": [
    "12119f3beb5b3328"
  ],
  "x": 595,
  "y": 320,
  "wires": [
    [
      "5e5f3ac7d04f4c40",
      "2282a0f6101a4056",
      "362fe99ed7ba2702",
      "58ec3a6c1cf472f2",
      "dd8012adb185c058",
      "4e0d14687b49f1e2",
      "bbf9c8ca9efb9fae"
    ]
  ]
}
```

```
},  
{  
  "id": "2282a0f6101a4056",  
  "type": "ui_gauge",  
  "z": "fb9a3a66b864403d",  
  "name": "",  
  "group": "e4f3410ae57e4461",  
  "order": 6,  
  "width": 0,  
  "height": 0,  
  "gtype": "gage",  
  "title": "Humidity",  
  "label": "%",  
  "format": "{{msg.payload.hum}}",  
  "min": 0,  
  "max": "100",  
  "colors": [  
    "#00b500",  
    "#e6e600",  
    "#ca3838"  
  ],  
  "seg1": "",  
  "seg2": "",  
  "diff": false,  
  "className": "",  
  "x": 740,  
  "y": 360,  
  "wires": []  
},  
{
```

```
"id": "362fe99ed7ba2702",
"type": "ui_text",
"z": "fb9a3a66b864403d",
"group": "129eb55ea17bdbce",
"order": 6,
"width": 0,
"height": 0,
"name": "",
"label": "Servo1 angle",
"format": "{{msg.payload.servo1}}",
"layout": "row-spread",
"className": "",
"style": false,
"font": "",
"fontSize": 16,
"color": "#000000",
"x": 750,
"y": 420,
"wires": []
},
{
  "id": "58ec3a6c1cf472f2",
  "type": "ui_text",
  "z": "fb9a3a66b864403d",
  "group": "129eb55ea17bdbce",
  "order": 7,
  "width": 0,
  "height": 0,
  "name": "",
  "label": "Servo2 angle",
```

```
"format": "{{msg.payload.servo2}}",
"layout": "row-spread",
"className": "",
"style": false,
"font": "",
"fontSize": 16,
"color": "#000000",
"x": 750,
"y": 480,
"wires": []
},
{
  "id": "dd8012adb185c058",
  "type": "ui_text",
  "z": "fb9a3a66b864403d",
  "group": "129eb55ea17bdbce",
  "order": 8,
  "width": 0,
  "height": 0,
  "name": "",
  "label": "Servo3 angle",
  "format": "{{msg.payload.servo3}}",
  "layout": "row-spread",
  "className": "",
  "style": false,
  "font": "",
  "fontSize": 16,
  "color": "#000000",
  "x": 750,
  "y": 540,
```

```
"wires": []
},
{
  "id": "0513832614dbb5a2",
  "type": "ui_slider",
  "z": "fb9a3a66b864403d",
  "name": "",
  "label": "Servo2",
  "tooltip": "",
  "group": "129eb55ea17bdbce",
  "order": 5,
  "width": 0,
  "height": 0,
  "passthru": true,
  "outs": "all",
  "topic": "topic",
  "topicType": "msg",
  "min": 0,
  "max": "180",
  "step": 1,
  "className": "",
  "x": 100,
  "y": 980,
  "wires": [
    [
      "badd7ac6025f038b"
    ]
  ]
},
{
```

```
"id": "805d2606e550c88a",
"type": "ui_slider",
"z": "fb9a3a66b864403d",
"name": "",
"label": "Servo3",
"tooltip": "",
"group": "129eb55ea17bdbce",
"order": 5,
"width": 0,
"height": 0,
"passthru": true,
"outs": "all",
"topic": "topic",
"topicType": "msg",
"min": 0,
"max": "180",
"step": 1,
"className": "",
"x": 100,
"y": 1040,
"wires": [
  [
    "2f00222e650864bf"
  ]
],
},
{
  "id": "ab39abd290e52525",
  "type": "function",
  "z": "fb9a3a66b864403d",
```

```

    "name": "servo1",
    "func": "msg.headers = { \"Content-Type\": \"application/json\"
};\nmsg.payload = {\n      \"servo1\": msg.payload\n};\nmsg.method =
\"POST\";\nmsg.url = \"http://192.168.0.101/command\";\nreturn msg;",
    "outputs": 1,
    "timeout": 0,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 290,
    "y": 920,
    "wires": [
      [
        "898ce854f961b4bc"
      ]
    ]
  },
  {
    "id": "badd7ac6025f038b",
    "type": "function",
    "z": "fb9a3a66b864403d",
    "name": "servo2",
    "func": "msg.headers = { \"Content-Type\": \"application/json\"
};\nmsg.payload = {\n      \"servo2\": msg.payload\n};\nmsg.method =
\"POST\";\nmsg.url = \"http://192.168.0.101/command\";\nreturn msg;",
    "outputs": 1,
    "timeout": 0,
    "noerr": 0,
    "initialize": "",

```

```

    "finalize": "",
    "libs": [],
    "x": 290,
    "y": 980,
    "wires": [
      [
        "4d33ae663376e393"
      ]
    ]
  },
  {
    "id": "2f00222e650864bf",
    "type": "function",
    "z": "fb9a3a66b864403d",
    "name": "servo3",
    "func": "msg.headers = { \"Content-Type\": \"application/json\"
};\nmsg.payload = {\n      \"servo3\": msg.payload\n};\nmsg.method =
\"POST\";\nmsg.url = \"http://192.168.0.101/command\"; \nreturn msg;",
    "outputs": 1,
    "timeout": 0,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 290,
    "y": 1040,
    "wires": [
      [
        "2c92b550f129513f"
      ]
    ]
  }
}

```

```
]
},
{
  "id": "4d33ae663376e393",
  "type": "http request",
  "z": "fb9a3a66b864403d",
  "name": "",
  "method": "use",
  "ret": "txt",
  "paytoqs": "ignore",
  "url": "http://192.168.1.104/command",
  "tls": "",
  "persist": false,
  "proxy": "",
  "insecureHTTPParser": false,
  "authType": "",
  "senderr": false,
  "headers": [],
  "x": 510,
  "y": 980,
  "wires": [
    [
      "a3f4fb8b74e589ae"
    ]
  ]
},
{
  "id": "a3f4fb8b74e589ae",
  "type": "debug",
  "z": "fb9a3a66b864403d",
```

```
"name": "debug 4",
"active": true,
"tosidebar": true,
"console": false,
"tostatus": false,
"complete": "false",
"statusVal": "",
"statusType": "auto",
"x": 680,
"y": 980,
"wires": []
},
{
  "id": "2c92b550f129513f",
  "type": "http request",
  "z": "fb9a3a66b864403d",
  "name": "",
  "method": "use",
  "ret": "txt",
  "paytoqs": "ignore",
  "url": "http://192.168.1.104/command",
  "tls": "",
  "persist": false,
  "proxy": "",
  "insecureHTTPParser": false,
  "authType": "",
  "senderr": false,
  "headers": [],
  "x": 510,
  "y": 1040,
```

```
"wires": [  
  [  
    "9c3b9aa9c1d396e2"  
  ]  
]  
},  
{  
  "id": "9c3b9aa9c1d396e2",  
  "type": "debug",  
  "z": "fb9a3a66b864403d",  
  "name": "debug 5",  
  "active": true,  
  "tosidebar": true,  
  "console": false,  
  "tostatus": false,  
  "complete": "false",  
  "statusVal": "",  
  "statusType": "auto",  
  "x": 680,  
  "y": 1040,  
  "wires": []  
},  
{  
  "id": "65cce656865d1e88",  
  "type": "inject",  
  "z": "fb9a3a66b864403d",  
  "name": "",  
  "props": [  
    {  
      "p": "payload"    }  
  ]  
}
```

```
    },  
    {  
      "p": "topic",  
      "vt": "str"  
    }  
  ],  
  "repeat": "",  
  "crontab": "",  
  "once": true,  
  "onceDelay": "0.5",  
  "topic": "",  
  "payload": "",  
  "payloadType": "date",  
  "x": 250,  
  "y": 760,  
  "wires": [  
    [  
      "f1787f4fe33fe0dc",  
      "7f796f780521fce6"  
    ]  
  ]  
},  
{  
  "id": "1f60f5024a12376c",  
  "type": "sqlite",  
  "z": "fb9a3a66b864403d",  
  "mydb": "4d437fe3b7e525cf",  
  "sqlquery": "msg.topic",  
  "sql": "",  
  "name": "",
```

```

"x": 830,
"y": 600,
"wires": [
  []
]
},
{
  "id": "f1787f4fe33fe0dc",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "create_db",
  "func": "msg.topic = `\\nCREATE TABLE IF NOT EXISTS telemetry (\\n
id INTEGER PRIMARY KEY AUTOINCREMENT,\\n  timestamp DATETIME
DEFAULT CURRENT_TIMESTAMP,\\n  temp REAL,\\n  hum REAL,\\n  servo1
INTEGER,\\n  servo2 INTEGER,\\n  servo3 INTEGER,\\n  distance_cm REAL,\\n
uptime_s INTEGER\\n);\\n`;\\nreturn msg;",
  "outputs": 1,
  "timeout": 0,
  "noerr": 0,
  "initialize": "",
  "finalize": "",
  "libs": [],
  "x": 440,
  "y": 760,
  "wires": [
    [
      "1f60f5024a12376c"
    ]
  ]
},

```

```

{
  "id": "a6ed352648ede64e",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "save_db",
  "func": "msg.timestamp = Date.now();\nmsg.topic = `INSERT INTO
telemetry (timestamp,temp, hum, servo1, servo2, servo3, distance_cm,
uptime_s)\nVALUES      (${msg.timestamp},      ${msg.payload.temp},
${msg.payload.hum},      ${msg.payload.servo1},      ${msg.payload.servo2},
${msg.payload.servo3},      ${msg.payload.distance_cm},
${msg.payload.uptime_s})`;
\nreturn msg;";
  "outputs": 1,
  "timeout": 0,
  "noerr": 0,
  "initialize": "",
  "finalize": "",
  "libs": [],
  "x": 480,
  "y": 540,
  "wires": [
    [
      "1f60f5024a12376c"
    ]
  ]
},
{
  "id": "4e0d14687b49fle2",
  "type": "ui_gauge",
  "z": "fb9a3a66b864403d",
  "name": ""
}

```

```
"group": "12a5bd7dbd9a8211",
"order": 10,
"width": 0,
"height": 0,
"gtype": "wave",
"title": "Distance",
"label": "units",
"format": "{{msg.payload.distance_cm}}",
"min": 0,
"max": "100",
"colors": [
  "#00b500",
  "#e6e600",
  "#ca3838"
],
"seg1": "",
"seg2": "",
"diff": false,
"className": "",
"x": 740,
"y": 240,
"wires": []
},
{
  "id": "25daab7752a9d477",
  "type": "ui_chart",
  "z": "fb9a3a66b864403d",
  "name": "",
  "group": "12a5bd7dbd9a8211",
  "order": 1,
```

```
"width": 0,  
"height": 0,  
"label": "chart",  
"chartType": "line",  
"legend": "true",  
"xformat": "HH:mm:ss",  
"interpolate": "linear",  
"nodata": "",  
"dot": false,  
"ymin": "0",  
"ymax": "100",  
"removeOlder": 1,  
"removeOlderPoints": "",  
"removeOlderUnit": "3600",  
"cutout": 0,  
"useOneColor": false,  
"useUTC": false,  
"colors": [  
    "#1f77b4",  
    "#aec7e8",  
    "#ff7f0e",  
    "#2ca02c",  
    "#98df8a",  
    "#d62728",  
    "#ff9896",  
    "#9467bd",  
    "#c5b0d5"  
],  
"outputs": 1,  
"useDifferentColor": false,
```

```

"className": "",
"x": 890,
"y": 160,
"wires": [
  []
]
},
{
  "id": "bbf9c8ca9efb9fae",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "chart",
  "func": "let temp = msg.payload.temp;\nlet hum = msg.payload.hum;\n\nlet
msgs = [\n  { topic: \"Температура\", payload: temp },\n  { topic: \"Вологість\",
payload: hum }\n];\n\nreturn [msgs];",
  "outputs": 1,
  "timeout": 0,
  "noerr": 0,
  "initialize": "",
  "finalize": "",
  "libs": [],
  "x": 730,
  "y": 160,
  "wires": [
    [
      "25daab7752a9d477"
    ]
  ]
},
{

```

```

    "id": "ebda825676e2e82a",
    "type": "function",
    "z": "fb9a3a66b864403d",
    "name": "alarms",
    "func": "let d = msg.payload;\n\nconst TEMP_HIGH = 30;\nconst
HUM_LOW = 40;\nconst WATER_LOW = 200;\nconst DIST_LOW = 10;\n\nif
(d.temp > TEMP_HIGH){\n  msg.payload.type = \"TEMP\";\n  msg.payload.value
= d.temp;\n  msg.payload.status = \"active\";\n}\nelse if (d.hum < HUM_LOW){\n
msg.payload.type = \"HUM\";\n  msg.payload.value = d.hum;\n  msg.payload.status
= \"active\";\n}\nelse{\n  msg.payload.type = \"NORMAL\";\n  msg.payload.value
= -1;\n      msg.payload.status    =  \"inactive\";\n}\n\nmsg.timestamp    =
Date.now();\n\nreturn msg;";
    "outputs": 1,
    "timeout": 0,
    "noerr": 0,
    "initialize": "",
    "finalize": "",
    "libs": [],
    "x": 110,
    "y": 660,
    "wires": [
      [
        "7c3d6d21dac8007e"
      ]
    ]
  },
  {
    "id": "7f796f780521fce6",
    "type": "function",
    "z": "fb9a3a66b864403d",

```

```

"name": "create_alarms",
"func": "msg.topic = `\\nCREATE TABLE IF NOT EXISTS alarms (\\n  id
INTEGER PRIMARY KEY AUTOINCREMENT,\\n  timestamp DATETIME
DEFAULT CURRENT_TIMESTAMP,\\n  type TEXT,\\n  value INTEGER,\\n  status
TEXT -- active / cleared\\n);\\n`;\\nreturn msg;",
"outputs": 1,
"timeout": 0,
"noerr": 0,
"initialize": "",
"finalize": "",
"libs": [],
"x": 440,
"y": 820,
"wires": [
  [
    "1f60f5024a12376c"
  ]
],
},
{
  "id": "7c3d6d21dac8007e",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "save_alarms",
  "func": "msg.topic = `\\nINSERT INTO alarms (timestamp, type, value,
status)\\nVALUES          (${msg.timestamp},          '${msg.payload.type}',
'${msg.payload.value}', '${msg.payload.status}')\\n`;\\nreturn msg;",
  "outputs": 1,
  "timeout": 0,
  "noerr": 0,

```

```
"initialize": "",
"finalize": "",
"libs": [],
"x": 330,
"y": 660,
"wires": [
  [
    "1f60f5024a12376c"
  ]
],
},
{
  "id": "d525ddf7022d5c73",
  "type": "ui_text_input",
  "z": "fb9a3a66b864403d",
  "name": "",
  "label": "time from",
  "tooltip": "",
  "group": "5c53a8148f2035e1",
  "order": 2,
  "width": 0,
  "height": 0,
  "passthru": true,
  "mode": "datetime-local",
  "delay": 300,
  "topic": "topic",
  "sendOnBlur": true,
  "className": "",
  "topicType": "msg",
  "x": 100,
```

```
"y": 1200,
"wires": [
  [
    "66ef4f6b5d8196f9"
  ]
],
{
  "id": "3a8cb552a33a54ce",
  "type": "ui_text_input",
  "z": "fb9a3a66b864403d",
  "name": "",
  "label": "time to",
  "tooltip": "",
  "group": "5c53a8148f2035e1",
  "order": 2,
  "width": 0,
  "height": 0,
  "passthru": true,
  "mode": "datetime-local",
  "delay": 300,
  "topic": "topic",
  "sendOnBlur": true,
  "className": "",
  "topicType": "msg",
  "x": 90,
  "y": 1240,
  "wires": [
    [
      "a8776e6ed0551e73"
```

```

    ]
  ]
},
{
  "id": "dd594a91b57594e5",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "function 6",
  "func": "let fromStr = flow.get(\"fromTime\");\nlet toStr =
flow.get(\"toTime\");\n\nif (!fromStr || !toStr) {\n  node.warn(\"Не задано часовий
проміжок!\");\n  return null;\n}\n\nlet fromTime = new Date(fromStr).getTime();\nlet
toTime = new Date(toStr).getTime();\n\nmsg.topic = `SELECT timestamp, type,
value, status\nFROM alarms\nWHERE timestamp BETWEEN ${fromTime} AND
${toTime}\nORDER BY timestamp ASC;\n`;\n\nreturn msg;",
  "outputs": 1,
  "timeout": 0,
  "noerr": 0,
  "initialize": "",
  "finalize": "",
  "libs": [],
  "x": 620,
  "y": 1260,
  "wires": [
    [
      "e52adebff2109ca2"
    ]
  ]
},
{
  "id": "8b1c3e7fa8225902",

```

```
"type": "ui_button",
"z": "fb9a3a66b864403d",
"name": "",
"group": "5c53a8148f2035e1",
"order": 4,
"width": 0,
"height": 0,
"passthru": false,
"label": "button",
"tooltip": "",
"color": "",
"bgcolor": "",
"className": "",
"icon": "",
"payload": "",
"payloadType": "str",
"topic": "topic",
"topicType": "msg",
"x": 470,
"y": 1260,
"wires": [
  [
    "dd594a91b57594e5"
  ]
]
},
{
  "id": "e52adebff2109ca2",
  "type": "sqlite",
  "z": "fb9a3a66b864403d",
```

```
"mydb": "4d437fe3b7e525cf",
"sqlquery": "msg.topic",
"sql": "",
"name": "",
"x": 830,
"y": 1180,
"wires": [
  [
    "e6bc570b0c7911bb",
    "efe2af4f25614e1f"
  ]
],
},
{
  "id": "a8776e6ed0551e73",
  "type": "change",
  "z": "fb9a3a66b864403d",
  "name": "",
  "rules": [
    {
      "t": "set",
      "p": "toTime",
      "pt": "flow",
      "to": "payload",
      "tot": "msg"
    }
  ],
  "action": "",
  "property": "",
  "from": "",
```

```
"to": "",
"reg": false,
"x": 300,
"y": 1240,
"wires": [
  []
],
},
{
  "id": "66ef4f6b5d8196f9",
  "type": "change",
  "z": "fb9a3a66b864403d",
  "name": "",
  "rules": [
    {
      "t": "set",
      "p": "fromTime",
      "pt": "flow",
      "to": "payload",
      "tot": "msg"
    }
  ],
  "action": "",
  "property": "",
  "from": "",
  "to": "",
  "reg": false,
  "x": 310,
  "y": 1200,
  "wires": [
```

```

    []
  ]
},
{
  "id": "e6bc570b0c7911bb",
  "type": "ui_table",
  "z": "fb9a3a66b864403d",
  "group": "5c53a8148f2035e1",
  "name": "",
  "order": 4,
  "width": "6",
  "height": "6",
  "columns": [],
  "outputs": 1,
  "cts": true,
  "x": 1110,
  "y": 1260,
  "wires": [
    []
  ]
},
{
  "id": "39165f96c1e6d2e2",
  "type": "function",
  "z": "fb9a3a66b864403d",
  "name": "filter",
  "func": "let alpha = flow.get(\"alpha_temp\") || 0.25;\nlet x =
msg.payload.temp;\nlet prev = flow.get(\"temp_filt\");\nif (prev === undefined) prev
= x;\nlet filt = alpha * x + (1 - alpha) * prev;\nflow.set(\"temp_filt\",

```

```
filt);\nmsg.payload.temp_filtered = Number(filt.toFixed(2));\nmsg.payload.prev =  
prev;\nreturn msg;";  
  "outputs": 1,  
  "timeout": 0,  
  "noerr": 0,  
  "initialize": "",  
  "finalize": "",  
  "libs": [],  
  "x": 350,  
  "y": 180,  
  "wires": [  
    [  
      "4478a6f4c61b334c",  
      "714c5fe964b8ceb9"  
    ]  
  ]  
},  
{  
  "id": "4478a6f4c61b334c",  
  "type": "debug",  
  "z": "fb9a3a66b864403d",  
  "name": "filter_temp",  
  "active": true,  
  "tosidebar": true,  
  "console": false,  
  "tostatus": false,  
  "complete": "payload.temp_filtered",  
  "targetType": "msg",  
  "statusVal": "",  
  "statusType": "auto",
```

```
"x": 530,  
"y": 180,  
"wires": []  
},  
{  
  "id": "714c5fe964b8ceb9",  
  "type": "debug",  
  "z": "fb9a3a66b864403d",  
  "name": "prev",  
  "active": true,  
  "tosidebar": true,  
  "console": false,  
  "tostatus": false,  
  "complete": "payload.prev",  
  "targetType": "msg",  
  "statusVal": "",  
  "statusType": "auto",  
  "x": 510,  
  "y": 120,  
  "wires": []  
},  
{  
  "id": "78bee7fd3626785f",  
  "type": "ui_text_input",  
  "z": "fb9a3a66b864403d",  
  "name": "",  
  "label": "time from",  
  "tooltip": "",  
  "group": "eda8d51b64e2f977",  
  "order": 2,
```

```
"width": 0,  
"height": 0,  
"passthru": true,  
"mode": "datetime-local",  
"delay": 300,  
"topic": "topic",  
"sendOnBlur": true,  
"className": "",  
"topicType": "msg",  
"x": 100,  
"y": 1400,  
"wires": [  
  [  
    "f3cde1bb669e0edf"  
  ]  
]  
},  
{  
  "id": "6f6f944f6883a4b6",  
  "type": "ui_text_input",  
  "z": "fb9a3a66b864403d",  
  "name": "",  
  "label": "time to",  
  "tooltip": "",  
  "group": "eda8d51b64e2f977",  
  "order": 2,  
  "width": 0,  
  "height": 0,  
  "passthru": true,  
  "mode": "datetime-local",
```

```

    "delay": 300,
    "topic": "topic",
    "sendOnBlur": true,
    "className": "",
    "topicType": "msg",
    "x": 90,
    "y": 1440,
    "wires": [
      [
        "c4d93943fb4a1a2d"
      ]
    ]
  },
  {
    "id": "99d38b91059b90bd",
    "type": "function",
    "z": "fb9a3a66b864403d",
    "name": "function 7",
    "func": "let fromStr1 = flow.get(\"fromTime1\").toString();\nlet toStr1 =
flow.get(\"toTime1\").toString();\n\n// if (!fromStr1 || !toStr1) {\n//   node.warn(\"He
задано часовий проміжок!\");\n//   return null;\n// }\n\n// // SQLite формат: YYYY-
MM-DD HH:MM:SS\n// let fromSQL = fromStr1.replace(\"T\", \" \") + \":00\";\n// let
toSQL = toStr1.replace(\"T\", \" \") + \":00\";\n\nmsg.topic = `SELECT timestamp,
temp\nFROM telemetry\nWHERE timestamp BETWEEN '${fromStr1}' AND
`${toStr1}`\nORDER BY timestamp ASC;\n`;\n\nreturn msg;";
    "outputs": 1,
    "timeout": 0,
    "noerr": 0,
    "initialize": "",
    "finalize": ""
  }

```

```
"libs": [],
"x": 620,
"y": 1460,
"wires": [
  [
    "6ab75c1531b4648c"
  ]
],
},
{
  "id": "1ca7c68d1509939b",
  "type": "ui_button",
  "z": "fb9a3a66b864403d",
  "name": "",
  "group": "eda8d51b64e2f977",
  "order": 4,
  "width": 0,
  "height": 0,
  "passthru": false,
  "label": "button",
  "tooltip": "",
  "color": "",
  "bgcolor": "",
  "className": "",
  "icon": "",
  "payload": "",
  "payloadType": "str",
  "topic": "topic",
  "topicType": "msg",
  "x": 470,
```

```
"y": 1460,
"wires": [
  [
    "99d38b91059b90bd"
  ]
],
},
{
  "id": "6ab75c1531b4648c",
  "type": "sqlite",
  "z": "fb9a3a66b864403d",
  "mydb": "4d437fe3b7e525cf",
  "sqlquery": "msg.topic",
  "sql": "",
  "name": "",
  "x": 830,
  "y": 1380,
  "wires": [
    [
      "e83aa5db532241e5",
      "ec73d060ecdf0f3b"
    ]
  ]
},
{
  "id": "c4d93943fb4a1a2d",
  "type": "change",
  "z": "fb9a3a66b864403d",
  "name": "",
  "rules": [
```

```
{
  "t": "set",
  "p": "toTime1",
  "pt": "flow",
  "to": "payload",
  "tot": "msg"
}
],
"action": "",
"property": "",
"from": "",
"to": "",
"reg": false,
"x": 300,
"y": 1440,
"wires": [
  []
]
},
{
  "id": "f3cde1bb669e0edf",
  "type": "change",
  "z": "fb9a3a66b864403d",
  "name": "",
  "rules": [
    {
      "t": "set",
      "p": "fromTime1",
      "pt": "flow",
      "to": "payload",
```

```

        "tot": "msg"
    }
],
"action": "",
"property": "",
"from": "",
"to": "",
"reg": false,
"x": 310,
"y": 1400,
"wires": [
    []
]
},
{
    "id": "e83aa5db532241e5",
    "type": "function",
    "z": "fb9a3a66b864403d",
    "name": "function 8",
    "func": "let alpha = 0.25;\nlet out = [];\nlet prev;\n\nfor (let i = 0; i <
msg.payload.length; i++) {\n    let row = msg.payload[i];\n    let x =
Number(row.temp);\n\n    if (isNaN(x)) continue;\n\n    if (prev === undefined) prev =
x;\n\n    let filt = alpha * x + (1 - alpha) * prev;\n\n    out.push({\n        timestamp:
row.timestamp,\n        temp_raw: x,\n        temp_filt: Number(filt.toFixed(2))\n    });\n\n    prev = filt;\n}\n\nmsg.payload = out;\nreturn msg;",
    "outputs": 1,
    "timeout": 0,
    "noerr": 0,
    "initialize": "",
    "finalize": "",

```

```
"libs": [],
"x": 1100,
"y": 1460,
"wires": [
  [
    "71aa6dc67cf0256e"
  ]
],
},
{
  "id": "71aa6dc67cf0256e",
  "type": "ui_table",
  "z": "fb9a3a66b864403d",
  "group": "eda8d51b64e2f977",
  "name": "",
  "order": 4,
  "width": "6",
  "height": "6",
  "columns": [],
  "outputs": 1,
  "cts": true,
  "x": 1250,
  "y": 1460,
  "wires": [
    []
  ]
},
{
  "id": "efe2af4f25614e1f",
  "type": "debug",
```

```
"z": "fb9a3a66b864403d",
"name": "test1",
"active": true,
"tosidebar": true,
"console": false,
"tostatus": false,
"complete": "payload",
"targetType": "msg",
"statusVal": "",
"statusType": "auto",
"x": 1190,
"y": 1180,
"wires": []
},
{
  "id": "ec73d060ecdf0f3b",
  "type": "debug",
  "z": "fb9a3a66b864403d",
  "name": "test2",
  "active": true,
  "tosidebar": true,
  "console": false,
  "tostatus": false,
  "complete": "payload",
  "targetType": "msg",
  "statusVal": "",
  "statusType": "auto",
  "x": 1210,
  "y": 1380,
  "wires": []
```

```
  },  
  {  
    "id": "129eb55ea17bdbce",  
    "type": "ui_group",  
    "name": "Control",  
    "tab": "e7365b9e71b46298",  
    "order": 1,  
    "disp": true,  
    "width": 6,  
    "collapse": false,  
    "className": ""  
  },  
  {  
    "id": "e4f3410ae57e4461",  
    "type": "ui_group",  
    "name": "Chart",  
    "tab": "e7365b9e71b46298",  
    "order": 2,  
    "disp": true,  
    "width": 6,  
    "collapse": false,  
    "className": ""  
  },  
  {  
    "id": "4d437fe3b7e525cf",  
    "type": "sqlitedb",  
    "db": "C:\\Users\\yurch\\OneDrive\\Рабочий  
стол\\magistr\\realization\\node-red\\test5.db",  
    "mode": "RWC"  
  },  
}
```

```
{
  "id": "12a5bd7dbd9a8211",
  "type": "ui_group",
  "name": "Chart Vizio",
  "tab": "e7365b9e71b46298",
  "order": 3,
  "disp": true,
  "width": 6,
  "collapse": false,
  "className": ""
},
{
  "id": "5c53a8148f2035e1",
  "type": "ui_group",
  "name": "Alerts",
  "tab": "bde4a9cd25bc342d",
  "order": 1,
  "disp": true,
  "width": 6,
  "collapse": false,
  "className": ""
},
{
  "id": "eda8d51b64e2f977",
  "type": "ui_group",
  "name": "Prediction",
  "tab": "62f7f548222842da",
  "order": 1,
  "disp": true,
  "width": 6,
```

```
"collapse": false,
"className": ""
},
{
  "id": "e7365b9e71b46298",
  "type": "ui_tab",
  "name": "Home",
  "icon": "dashboard",
  "disabled": false,
  "hidden": false
},
{
  "id": "bde4a9cd25bc342d",
  "type": "ui_tab",
  "name": "Alerts",
  "icon": "dashboard",
  "order": 2,
  "disabled": false,
  "hidden": false
},
{
  "id": "62f7f548222842da",
  "type": "ui_tab",
  "name": "Prediction",
  "icon": "dashboard",
  "disabled": false,
  "hidden": false
},
{
  "id": "faf2a921e3ca31f2",
```

```
"type": "global-config",  
"env": [],  
"modules": {  
  "node-red-dashboard": "3.6.6",  
  "node-red-node-sqlite": "1.1.1",  
  "node-red-node-ui-table": "0.4.5"  
}  
}  
]
```

ДОДАТОК Г

Демонстраційний матеріал

Міністерство освіти та науки України
Харківський національний університет радіоелектроніки

Кафедра КІТАР
КВАЛІФІКАЦІЙНА РОБОТА

На тему: Розроблення SCADA-системи керування виробничою лінією з використанням IoT

Виконав:
студент групи КТРСм-24-1
Юрченко Олег Дмитрович

Керівник:
доц. каф. КІТАР
Бронніков Артем Ігорович

Відомості роботи

Мета кваліфікаційної роботи – підвищення ефективності спостереження, передачі даних з датчиків виробничої лінії для гнучкого коригування та аналізу виробничого процесу.

Об'єкт дослідження – процес управління, контролю та збору даних з датчиків виробничої лінії та передачі даних для подальшого аналізу та відображення.

Предмет дослідження – моделі та технології для системи контролю та збору даних на базі HTTP протоколу з використанням технологій IoT для виробничої лінії.

Огляд SCADA-систем

Основні функції:

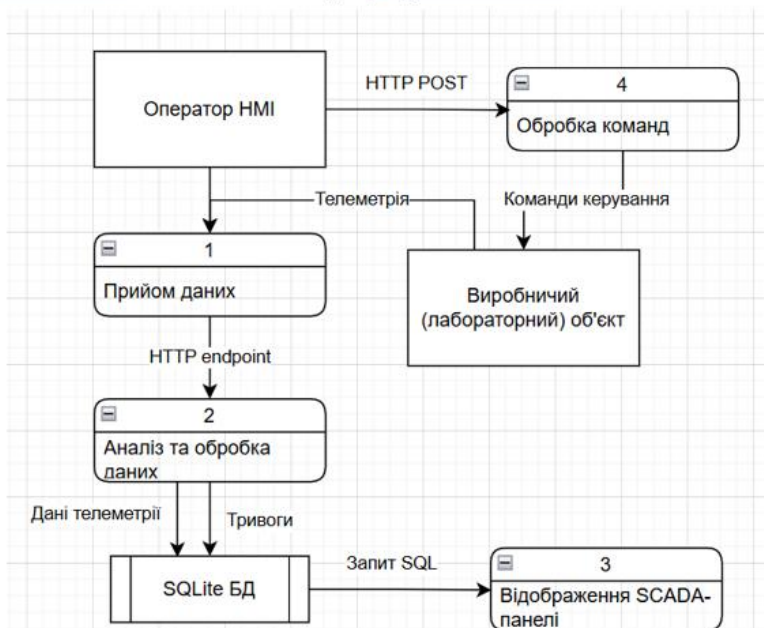
- збір даних;
- HMI;
- аналіз та обробка даних;
- створення звітності;
- збереження історичних даних;
- режим реального часу;
- сигналізація тривоги.



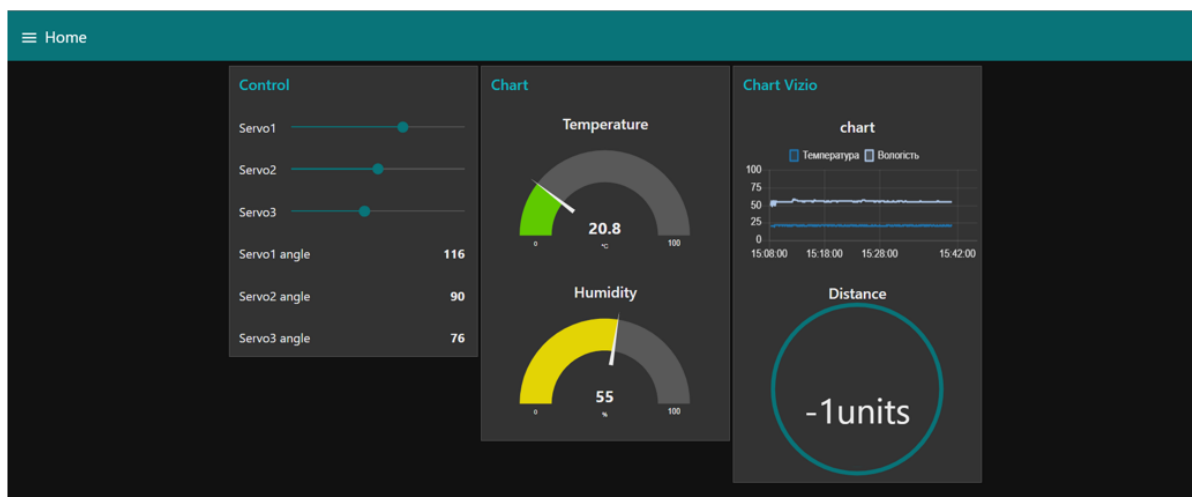
Структура базової SCADA-системи



DFD-діаграма потоку даних



Розроблена SCADA-система



Alert

time from
07.12.2025 12:00

time to
07.12.2025 15:30

BUTTON

tim...	type	value	status
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive
1765112...	NORMAL	-1	inactive

Збереження даних

```
CREATE TABLE IF NOT EXISTS telemetry (
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  timestamp DATETIME DEFAULT CURRENT_TIMESTAMP,
  temp REAL,
  hum REAL,
  servo1 INTEGER,
  servo2 INTEGER,
  servo3 INTEGER,
  distance_cm REAL,
  uptime_s INTEGER);
```

Таблиця збереження даних

```
CREATE TABLE IF NOT EXISTS alarms (
  id INTEGER PRIMARY KEY AUTOINCREMENT,
  timestamp DATETIME DEFAULT CURRENT_TIMESTAMP,
  type TEXT,
  value INTEGER,
  status TEXT -- active / cleared);
```

Таблиця збереження тривоги

$$Alarm = \begin{cases} TEMP_HIGH, & T > T_{max}, \\ HUM_LOW, & H > H_{min}, \\ DIST_SHORT, & H > H_{min}, \\ NORMAL, & \text{інше.} \end{cases}$$

Алгоритм виявлення тривоги

Експоненціальне згладжування

$$\hat{x}_t = \alpha x_t + (1 - \alpha)\hat{x}_{t-1},$$

$\alpha \in [0; 1]$ – коефіцієнт згладжування.

```

10.12.2025, 21:30:55 node: debug 2
telemetry : msg.payload : Object
  { temp: 20.4, hum: 46, servo1: 90, servo2: 90, servo3: 90 ... }

10.12.2025, 21:30:56 node: filter_temp
telemetry : msg.payload.temp_filtered : number
20.41

10.12.2025, 21:30:56 node: prev
telemetry : msg.payload.prev : number
20.41777830078609
  
```

Онлайн згладжування поточного значення на основі попереднього

```

Properties
Name: filter
Setup | On Start | On Message | On Stop
1 let alpha = flow.get("alpha_temp") || 0.25;
2 let x = msg.payload.temp;
3 let prev = flow.get("temp_filt");
4 if (prev === undefined) prev = x;
5 let filt = alpha * x + (1 - alpha) * prev;
6 flow.set("temp_filt", filt);
7 msg.payload.temp_filtered = Number(filt.toFixed(2));
8 msg.payload.prev = prev;
9 return msg;
  
```

Програмна реалізація методу згладжування

Пакетна фільтрація даних з БД

Prediction

Prediction

time from: 10.12.2025 00:28

time to: 10.12.2025 21:28

BUTTON

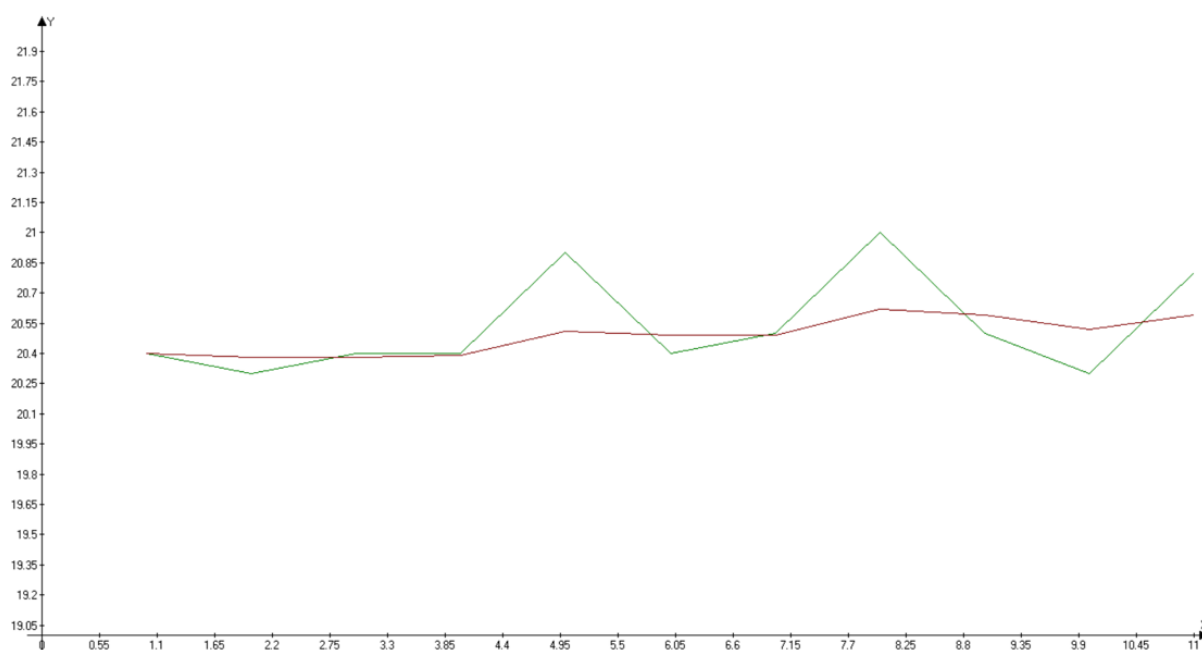
timesta...	temp_raw	temp_filt
1765392014...	20.2	20.2
1765392019...	20.2	20.2
1765392024...	20.2	20.2
1765392029...	20.2	20.2
1765392034...	20.2	20.2
1765392039...	20.2	20.2
1765392044...	20.2	20.2
1765392049...	20.2	20.2
1765392054...	20.2	20.2
1765392059...	20.2	20.2
1765392064...	20.2	20.2

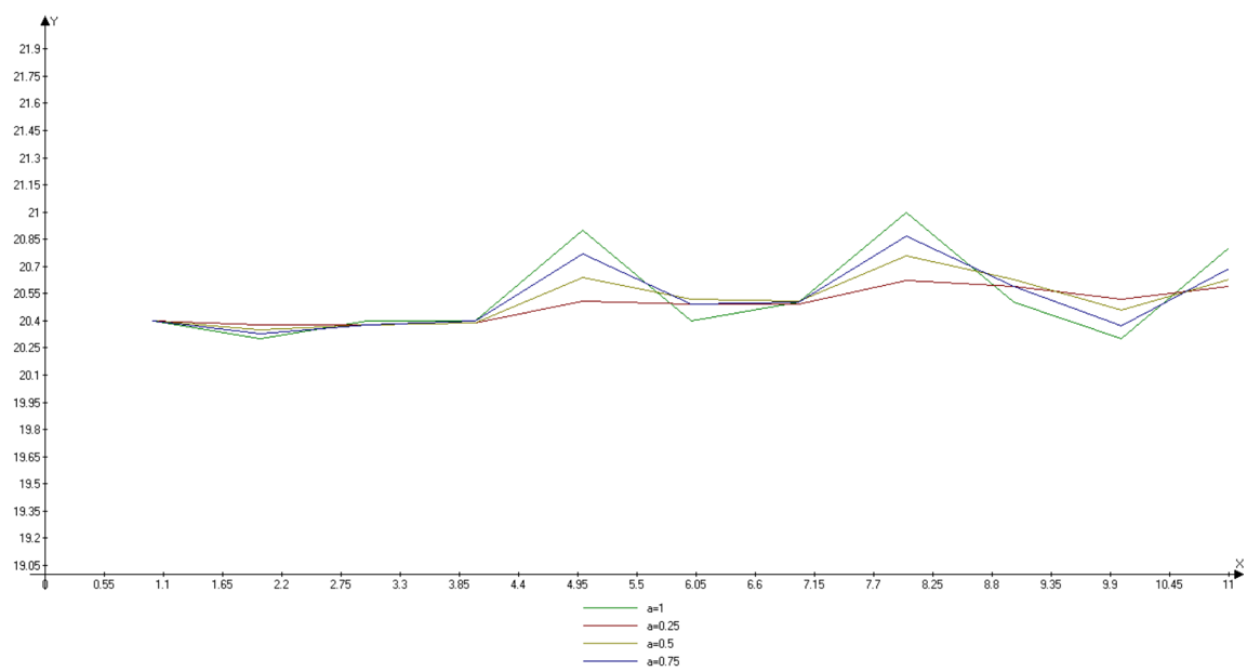
Дослідження експоненціального згладжування

Часова мітка	Згладжені дані	Початкові дані
1765475636892	20.4	20.4
1765475641910	20.38	20.3
1765475646882	20.38	20.4
1765475651886	20.39	20.4
1765475656884	20.51	20.9
1765475661867	20.49	20.4
1765475666998	20.49	20.5
1765475671897	20.62	21
1765475676932	20.59	20.5
1765475681901	20.52	20.3
1765475686891	20.59	20.8

σ 0,0855 0,2346

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2}$$





Прогнозування значень температури

$$x_{t+1} = x_t + \Delta_t.$$

Часова мітка		Значення температури	Δ_t
1765475646882		20.4	–
1765475651886		20.38	-0,02
1765475656884		20.38	0
1765475661867		20.39	0,01
Кроки	Прогноз	Фактичне	Похибка
1	20,40	20,51	0,11°C
2	20,41	20,49	0,08°C
3	20,42	20,49	0,07°C
5	20,44	20,59	0,15°C

ВИСНОВКИ

У підсумку розроблена система виконує всі ключові функції SCADA-рівня:

- збір телеметрії;
- фільтрація та аналіз даних;
- виявлення тривоги;
- керування виконавчими механізмами;
- збереження історичних даних;
- побудова графіків та аналітики;
- функціональність цифрового двійника.

№ доку-мента	Позначення	Найменування	Додаткові відомості
		<u>Текстові документи</u>	
1	ГЮИК. 505111.013 ПЗ	Пояснювальна записка	A4, 145 с.
		<u>Додаткові матеріали</u>	
2		Результати роботи у статті	A4, 9 с.
3		Програмна реалізація для ESP32	A4, 5 с.
4		Програмна реалізація для Node-RED	A4, 46 с.
5		Демонстраційний матеріал у вигляді презентації	A4, 17 с.
		ГЮИК.505111.013 ВД	
Змін.	Арк.	Номер докум.	Підпис
			Дата
Розроб.	Юрченко О.Д.		
Перевір.	Бронніков А.І.		
Н.контр.	Демська Н.П.		
Затв.	Невлюдов І.Ш.		
		Розроблення SCADA-системи керування виробничою лінією з використанням IoT	Літера Н
		Відомість кваліфікаційної роботи	Аркуш 1
			Аркушів 1
			Кафедра КІТАР
			ХНУРЕ