

Не містить відомостей заборонених до відкритого публікування

Здобувач _____ / Кирило Сердюков /
(підпис) (прізвище та ініціали)

Керівник _____ / Олександр Бибка /
(підпис) (прізвище та ініціали)

Харківський національний університет радіоелектроніки

Факультет _____ інфокомунікацій _____
Кафедра _____ інформаційно-мережної інженерії _____
Рівень вищої освіти _____ перший (бакалаврський) _____
Спеціальність _____ 172 Телекомунікації та радіотехніка _____
(код і повна назва)
Тип програми _____ освітньо-професійна _____
(освітньо-професійна або освітньо-наукова)
Освітня програма _____ інформаційно-мережна інженерія _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« _____ » _____ 20 ____ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Сердюкову Кирилу Вячеславовичу _____
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка електронного годинника на базі Arduino

затверджена наказом університету від 23 травня 2025 р. № 410 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 24 червня 2025 р.

3. Вихідні дані до роботи _____

Провести аналіз особливостей онлайн та офлайн симулятори Arduino.

Розробити прототип електронного годинника на базі плати Arduino з додатковими функціями.

4. Перелік питань, що потрібно опрацювати в роботі _____

Вступ

1. Онлайн та офлайн симулятори Arduino

2. Розробка електронного годинника

Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
Слайди у форматі Power Point (назва та мета роботи, онлайн та офлайн симулятори, компоненти, симуляція, підтвердження працездатності, висновки)

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Ознайомлення із завданням. Уточнення ТЗ.	23.05.2025	вик.
2	Підбір літератури за темою роботи	24.05.-28.05.2025	вик.
3	Онлайн та офлайн симулятори Arduino	29.05.-02.06.2025	вик.
4	Розробка електронного годинника	03.06.2024-19.06.2025	вик.
5	Оформлення презентаційного матеріалу, підготовка до захисту у ЕК	20.06.-21.06.2025	вик.

Дата видачі завдання 23 травня 2025 р.

Здобувач _____ Кирило Сердюков
(підпис)

Керівник роботи _____ ст. викл. Олександр Бибка
(підпис) (посада, власне ім'я, прізвище)

РЕФЕРАТ

Пояснювальна записка: 48 с., 23 рис., 2 табл., 11 джерел, 2 додатки.

Мета роботи – розробити та реалізувати прототип електронного годинника з використанням платформи Arduino, який буде відображати поточний час та за потреби додаткові функції (наприклад, будильник, температура, календар тощо).

Об'єкт дослідження – процес проєктування, програмування та апаратної реалізації електронного годинника з використанням мікроконтролера Arduino.

Симулятори бувають онлайн та офлайн, кожен з яких має свої особливості. Онлайн-симулятори (наприклад, Wokwi або Tinkercad Circuits) працюють прямо в браузері, не потребують встановлення, мають інтуїтивно зрозумілий інтерфейс та зручні для швидкого доступу з будь-якого пристрою. Вони ідеально підходять для дистанційного навчання, експериментів, спільної роботи та ознайомлення з основами. Офлайн-симулятори (як-от Proteus або Fritzing) забезпечують глибший функціонал, часто підтримують моделювання складніших схем, точнішу роботу аналогових компонентів і не залежать від інтернет-з'єднання.

СИМУЛЯТОР, ПЛАТА, ARDUINO, МАТРИЧНИЙ ДИСПЛЕЙ,
МОДУЛЬ РЕАЛЬНОГО ЧАСУ, ДАТЧИК

THE ABSTRACT

Explanatory note: 48 p., 23 fig, 2 tabl., 11 sources, 2 app.

The purpose of the work is to develop and implement a prototype of an electronic clock using the Arduino platform, which will display the current time and, if necessary, additional functions (for example, alarm clock, temperature, calendar, etc.).

The object of research is the process of designing, programming and hardware implementation of an electronic clock using the Arduino microcontroller.

Simulators come in both online and offline versions, each with its own unique features. Online simulators (such as Wokwi or Tinkercad Circuits) run directly in your browser, require no installation, have an intuitive interface, and are easy to access from any device. They are ideal for distance learning, experimentation, collaboration, and learning the basics. Offline simulators (such as Proteus or Fritzing) provide deeper functionality, often support more complex circuit simulations, more accurate analog component operation, and are not dependent on an internet connection.

SIMULATOR, BOARD, ARDUINO, MATRIX DISPLAY, REAL TIME
MODULE, SENSOR

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	8
ВСТУП.....	9
1 ОНЛАЙН ТА ОФЛАЙН СИМУЛЯТОРИ ARDUINO.....	10
1.1 Основні відомості	10
1.2 Tinkercad	11
1.3 Proteus.....	12
1.4 Virtual Breadboard.....	14
1.5 Autodesk EAGLE	14
1.6 Arduino IO Simulator Drag, Drop & Draw	16
1.7 Arduino IDE.....	17
1.8 Wokwi.....	19
2 РОЗРОБКА ЕЛЕКТРОННОГО ГОДИННИКА.....	22
2.1 Мета проєкту	22
2.2 Компоненти	22
2.3 Симуляція	29
2.4 Живлення	37
ВИСНОВКИ.....	40
ПЕРЕЛІК ПОСИЛАНЬ	41
ДОДАТОК А	42
ДОДАТОК Б.....	43

ПЕРЕЛІК СКОРОЧЕНЬ

Arduino IDE – Arduino Integrated Development Environment – інтегроване середовище розробки для Windows, macOS та Linux, розроблене на Cі та C++;

HMI – Human Machine Interface – інтерфейс «людина-машина»;

HTTP – Hypertext Transfer Protocol – протокол передачі гіпертексту;

LCD – Liquid Crystal Display – рідкокристалічний екран;

MQTT – Message Queue Telemetry Transport – відкритий протокол обміну повідомленнями;

NTP – Network Time Protocol – протокол мережного часу;

OLED – Organic Light-Emitting Display – органічний світлодіодний дисплей;

OS – Operating System – операційна система;

SPI – Serial Peripheral Interface – серійний периферійний інтерфейс;

ULP – User Language Programs – користувацькі мовні програми;

USB – Universal Serial Bus – універсальна послідовна шина;

Wi-Fi – Wireless Fidelity – безпроводова мережа;

ПК – персональний комп'ютер.

ВСТУП

У сучасному світі розробка електроніки стає доступнішою завдяки використанню симуляторів мікроконтролерів, зокрема Arduino. Для навчання, тестування та прототипування проектів дедалі частіше застосовуються онлайн та офлайн симулятори, які дозволяють працювати з віртуальними електронними компонентами без необхідності у фізичному обладнанні. Вони дають можливість писати код, створювати схеми, аналізувати їхню роботу та усувати помилки ще до етапу реального збирання. Завдяки цьому симулятори стали незамінними інструментами як у сфері освіти, так і в професійній розробці. Саме тому з'явилися симулятори Arduino — програми, що імітують роботу апаратного забезпечення та дозволяють користувачам створювати віртуальні електронні схеми.

Симулятори бувають онлайн та офлайн, кожен з яких має свої особливості. Онлайн-симулятори (наприклад, Wokwi або Tinkercad Circuits) працюють прямо в браузері, не потребують встановлення, мають інтуїтивно зрозумілий інтерфейс та зручні для швидкого доступу з будь-якого пристрою. Вони ідеально підходять для дистанційного навчання, експериментів, спільної роботи та ознайомлення з основами. Офлайн-симулятори (як-от Proteus або Fritzing) забезпечують глибший функціонал, часто підтримують моделювання складніших схем, точнішу роботу аналогових компонентів і не залежать від інтернет-з'єднання.

Завдяки цим інструментам, розробники можуть швидко виявляти помилки, вивчати поведінку електронних схем у різних умовах та створювати повноцінні прототипи ще до виготовлення реального пристрою. Онлайн та офлайн симулятори Arduino стали потужними засобами для підвищення ефективності навчального процесу, пришвидшення розробки та зниження витрат на матеріали.

1 ОНЛАЙН ТА ОФЛАЙН СИМУЛЯТОРИ ARDUINO

1.1 Основні відомості

Моделювання подій в реальному часі було основою багатьох галузей. Протягом багатьох років деякі великі процеси симуляції були областю авіонавтики і авіації. Сьогодні симулятори Ардуіно дозволяють всім новачкам і професійним проектувальникам вчитися програмувати і тестувати ідеї, не побоюючись втрати енергії даремно разом зі своїми грошима [1].

Симулятор Arduino - відмінні платформи для програмістів і дизайнерів, які хочуть вивчити основи проектування та схемотехніки. Успіх таких програм пов'язаний з тим, що він надає вам можливість вчитися, не побоюючись пошкодити пристрій. Крім того, учні у яких можуть виникнути проблеми з придбанням електричного устаткування, не маючи поняття про те, як вони будуть функціонувати, можуть зрозуміти багато нюансів через проби і помилки за допомогою цих симуляторів. Це заощадить вам багато грошей і часу. Ще одна велика перевага симуляторів Ардуіно полягає в тому, що він підтримує порядково налагодження, тому користувач точно знає, де і в якому рядку він або вона зробив щось не так [1].

Чи можна займатися Ардуіно проектами без самого Ардуіно? Виявляється, цілком. Завдяки новому сервісу Tinkercad Circuits Arduino можна не тільки малювати електронні схеми і запускати емулятор електричного кола, а й підключати до проекту віртуальну плату Ардуіно, в яку можна завантажити справжні скетчі. Тінкеркад - безкоштовний, дивно простий і одночасно потужний сервіс, з якого можна починати навчання електроніці та робототехніці. Не потрібно нічого качати, все доступно онлайн [1].

Симулятором називають пристрій або сервіс, що імітують певні функції іншої системи, але не претендують на створення точної копії. Це деяка віртуальне середовище, в якій ми просто моделюємо іншу систему. Емулятор -

це повноцінний аналог, здатний замінити оригінал. Tinkercad симулює роботу електронних схем і контролера, але при цьому він є емулятором Ардуіно, реалізуючи практично всі базові функції Arduino IDE - від середовища редагування і компілятора до монітора порту і підключення бібліотек [1].

1.2 Tinkercad

Tinkercad - це онлайн сервіс, який зараз належить мастодонтів світу CAD-систем - компанії Autodesk. Тінкеркад вже давно відомий багатьом як просте і безкоштовне середовище для навчання 3D-моделювання. З її допомогою можна досить легко створювати свої моделі і відправляти їх на 3D-друк. Зовсім недавно Тінкеркад отримав можливість створення електронних схем і підключення їх до симулятора віртуальної плати Ардуіно. Ці вкрай важливі і потужні інструменти здатні істотно полегшити процеси навчання, проектування та програмування нових схем початківцям розробникам Arduino [1].

Tinkercad дозволяє створювати та тестувати свої проекти. Він має простий у використанні інтерфейс та велику бібліотеку компонентів, які ви можете використовувати у своїх проектах. Ви можете перетягувати та розміщувати компоненти на віртуальній дошці Arduino, а потім програмувати їх за допомогою мови Arduino. Tinkercad також має можливість візуалізації вихідних даних, що дозволяє вам перевірити, чи працює ваш проект насправді.

Можливості симулятора Tinkercad для розробника Arduino [1]:

- онлайн платформа, для роботи не потрібно нічого крім браузера і стійкого інтернету;
- зручний графічний редактор для візуальної побудови електронних схем;
- набір попередньо встановлених моделей більшості популярних електронних компонентів, відсортоване за типами компонентів;

- симулятор електронних схем, за допомогою якого можна підключити створений віртуальний пристрій до віртуального джерела живлення і простежити, як воно буде працювати;
- симулятор датчиків та інструментів зовнішнього впливу. Ви можете змінювати показання датчиків, стежачи за тим, як на них реагує система;
- вбудований редактор Arduino з монітором порту і можливістю покрокового налагодження;
- готові для розгортання проекти Arduino зі схемами і кодом;
- візуальний редактор коду Arduino;
- вбудовані підручники і величезне співтовариство з колекцією готових проектів.

Не потрібно завантажувати Arduino IDE, не потрібно шукати і викачувати популярні бібліотеки і скетчі, не потрібно збирати схему і підключати плату - все, що нам потрібно, знаходиться відразу на одній сторінці. І, найголовніше - це все дійсно працює [1].

Для початку роботи необхідно отримати обліковий запис Autocad. Реєстрація в Tinkercad абсолютно безкоштовна. Зайдіть на сайт і виконайте прості кроки. Підтвердивши аккаунт поштою, увійдіть в систему, вказавши введені параметри. У верхньому правому куті ви побачите посилання в особистий кабінет. У режимі редагування профілю ви зможете поміняти свій псевдонім, email, опис, встановити фотографію [1].

1.3 Proteus

Proteus — це програмне забезпечення для електроніки, за допомогою якого можна створювати та моделювати електронні схеми. Крім того, у вас є редактор для смартфона, щоб керувати всіма своїми проектами дистанційно за допомогою плат Arduino або Raspberry Pi [2].

Proteus – це офлайн-симулятор Arduino, який має велику бібліотеку компонентів, які можна використовувати у своїх проєкт. Можна перетягувати

та розміщувати компоненти на віртуальній дошці Arduino, а потім програмувати їх за допомогою мови Arduino. Proteus (рис. 1.1) також має можливість візуалізації вихідних даних, що дозволяє перевірити, чи працює проект насправді [2].

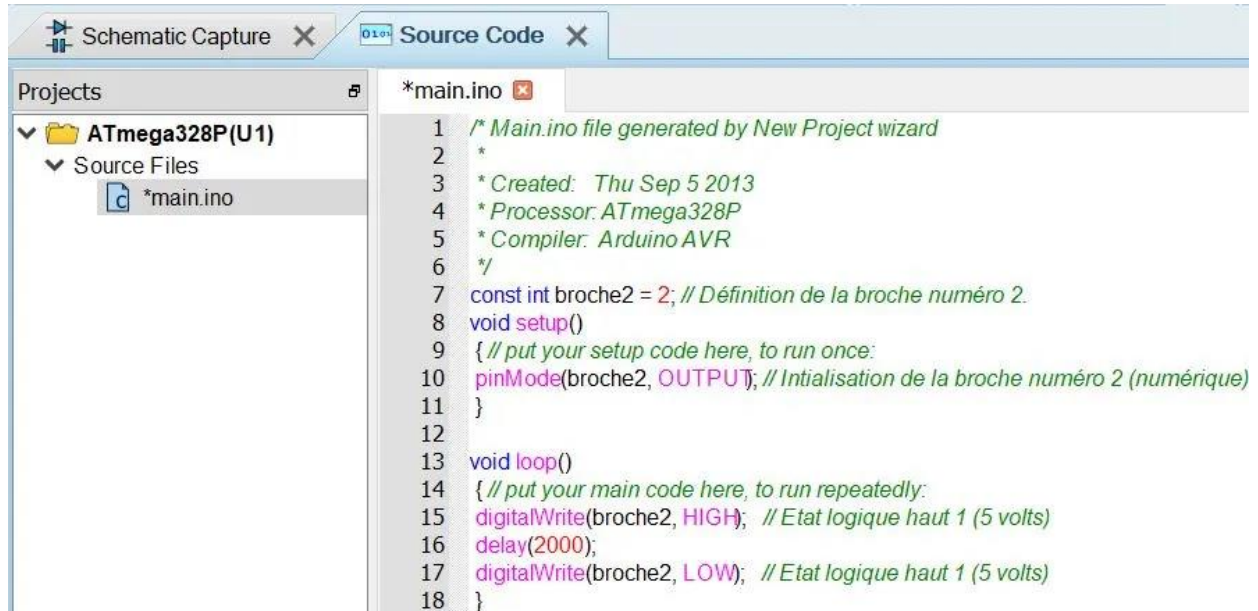


Рисунок 1.1 – Робоче вікно Proteus [3]

Proteus – це справді гарне програмне забезпечення для створення та моделювання ваших схем. Різниця між Proteus Tinkercad, Fritzing та Flowcode полягає в тому, що Proteus менше використовується для карт Arduino та Raspberry Pi. Він більше підходить для електронних карт, таких як Microchip [2].

Крім того, Proteus має більше опцій, ніж інше програмне забезпечення для електроніки, таких як рахунок вартості вашої схеми або вимірювальні прилади.

Таким чином, Proteus — це справді гарне програмне забезпечення, яке більше використовується для професійних цілей. Якщо ви новачок в електроніці, ми рекомендуємо вам використовувати програмне забезпечення Tinkercad для створення та моделювання вашої схеми [2].

1.4 Virtual Breadboard

Virtual Breadboard – це офлайн-симулятор Arduino, який дозволяє моделювати та тестувати свої проекти на комп'ютері. Він має велику бібліотеку компонентів, які ви можете використовувати у своїх проектах. Ви можете перетягувати та розміщувати компоненти на віртуальній дошці Arduino, а потім програмувати їх за допомогою мови Arduino. Virtual Breadboard (рис. 1.2) також має можливість візуалізації вихідних даних, що дозволяє вам перевірити, чи працює ваш проект насправді [3].

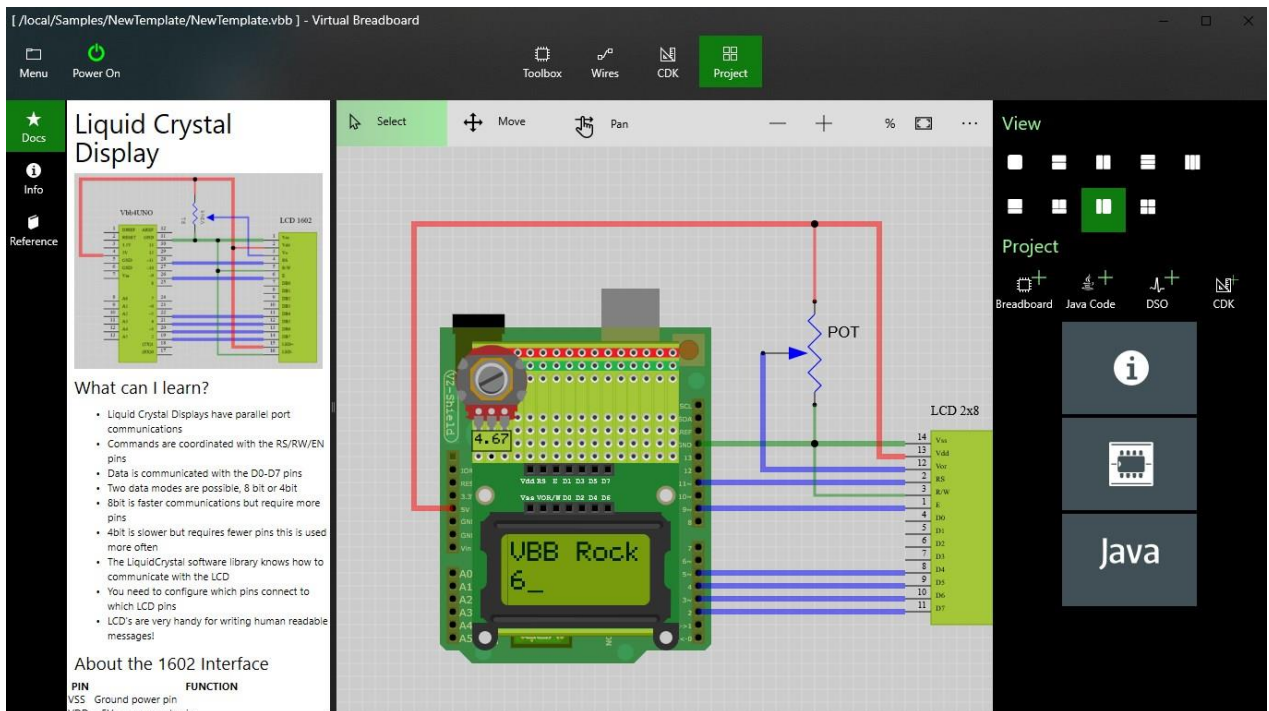


Рисунок 1.2 – Робоче вікно Virtual Breadboard [3]

1.5 Autodesk EAGLE

Autodesk EAGLE – це офлайн-симулятор Arduino, який дозволяє моделювати та тестувати свої проекти на комп'ютері. Він має велику бібліотеку компонентів, які ви можете використовувати у своїх проектах. Ви можете перетягувати та розміщувати компоненти на віртуальній дошці Arduino, а потім

програмувати їх за допомогою мови Arduino. Autodesk EAGLE (рис. 1.3) також має можливість візуалізації вихідних даних, що дозволяє вам перевірити, чи працює ваш проект насправді [3].

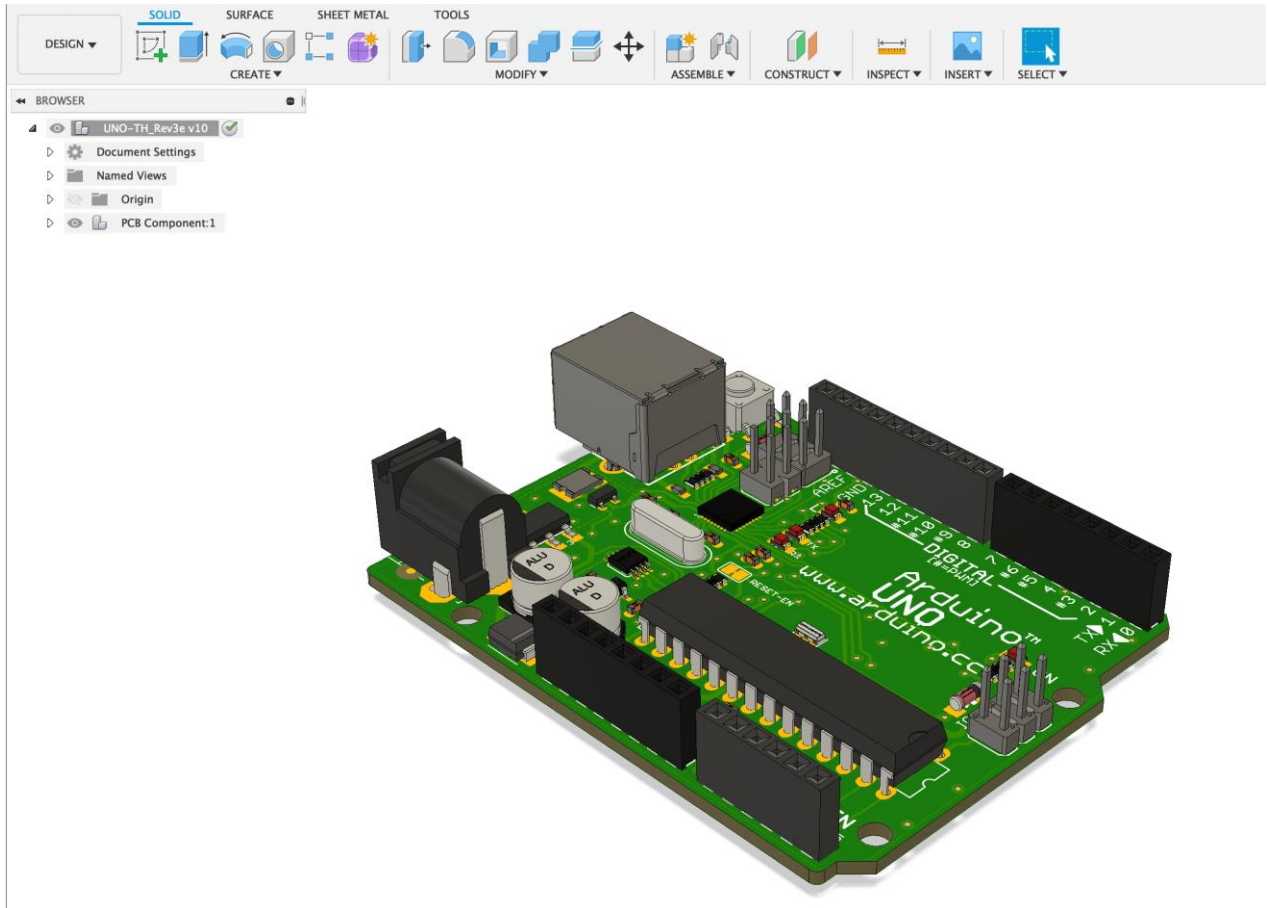


Рисунок 1.3 – Робоче вікно Virtual Breadboard [3]

Особливості Eagle:

- схематичний редактор;
- модульна конструкція блоків;
- перевірка електричних правил;
- редактор макетів друкованих плат BGA fanout;
- високошвидкісне проектування 3D-моделі для розміщення друкованих плат;
- користувацькі мовні програми (ULP).

1.6 Arduino IO Simulator Drag, Drop & Draw

Arduino IO Simulator Drag, Drop & Draw (рис. 1.4) – це симулятор Arduino, який дозволяє визначитись як буде працювати який-небудь датчик чи світлодіод чи інший компонент з платою, та програмою перед тим, як його придбати [4].

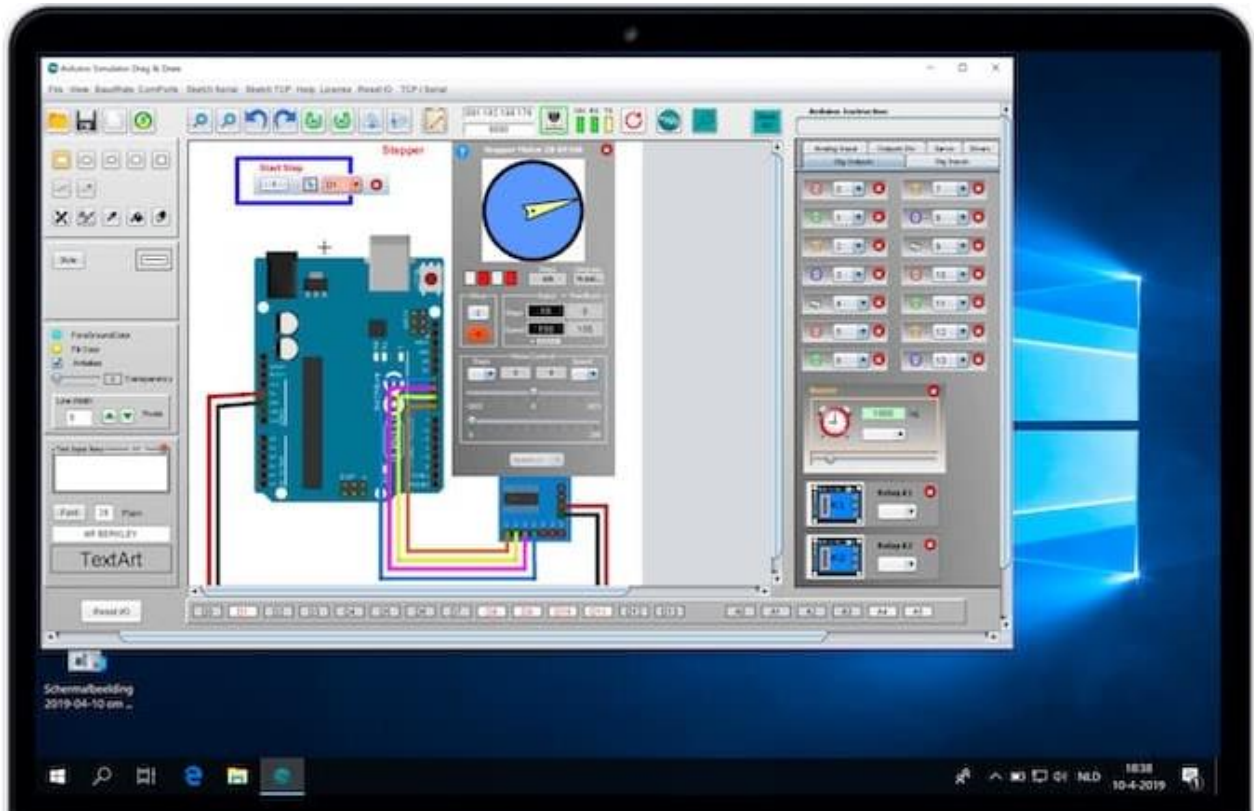


Рисунок 1.4 – Робоче вікно Arduino IO Simulator Drag, Drop & Draw [4]

Є можливість перетягувати мишкою компоненти на робоче поле та малювати все що завгодно на ньому. Також можна комбінувати порти вводу-виводу з комп'ютером чи Arduino, тобто ви можете створити інтерфейс «людина-машина» (HMI, Human Machine Interface) [4].

Симулятор має багато компонентів для проекту і їх список постійно оновлюється, а якщо вам потрібен власний особливий сенсор, ви можете створити його самі [4].

Arduino IO Simulator працює з наступними платами [4]:

- Arduino UNO;
- Arduino Mega;
- Arduino Nano;
- Arduino Micro;
- Arduino Leonardo.

Програма дозволяє малювати кола, квадрати, стрілки, прямокутники та інші фігури любого кольору. Також на робочому полі можна розміщувати текст [4].

1.7 Arduino IDE

Arduino IDE (Integrated Development Environment) – це офіційне середовище для розробки програмного забезпечення для мікроконтролерів платформи Arduino. Воно розроблене для спрощення написання, компіляції, завантаження та тестування коду на мікроконтролерах. Arduino IDE ідеально підходить як для початківців, так і для досвідчених розробників [5].

Arduino IDE призначене для створення та редагування програмного коду для плат, на базі таких мікроконтролерів як: ATmega328, ATmega32U4, ATmega2560, AT91SAM3X8E, SAMD21, ESP8266, Intel x86, Attiny85, STM [5].

Додаток Arduino IDE завоював популярність серед багатьох користувачів завдяки своїй простоті, зручності та багатому набору готових бібліотек та підтримуваних мікроконтролерів, список яких постійно зростає [5].

За допомогою Arduino IDE створюють скетчі як для простих мигалок, таймерів та перемикачів навантажень, так і для систем розумного будинку, верстатів з ЧПУ та квадрокоптерів. З Arduino IDE проекти обмежуються лише фантазією [5].

У програмуванні використовується спрощена версія мови C++, що забезпечує легкість і простоту в освоєнні для програмістів-початківців електронників [5].

На офіційному сайті arduino.cc є кілька версій Arduino IDE (рис. 1.5) для операційних систем Windows, Mac OS та Linux [5].

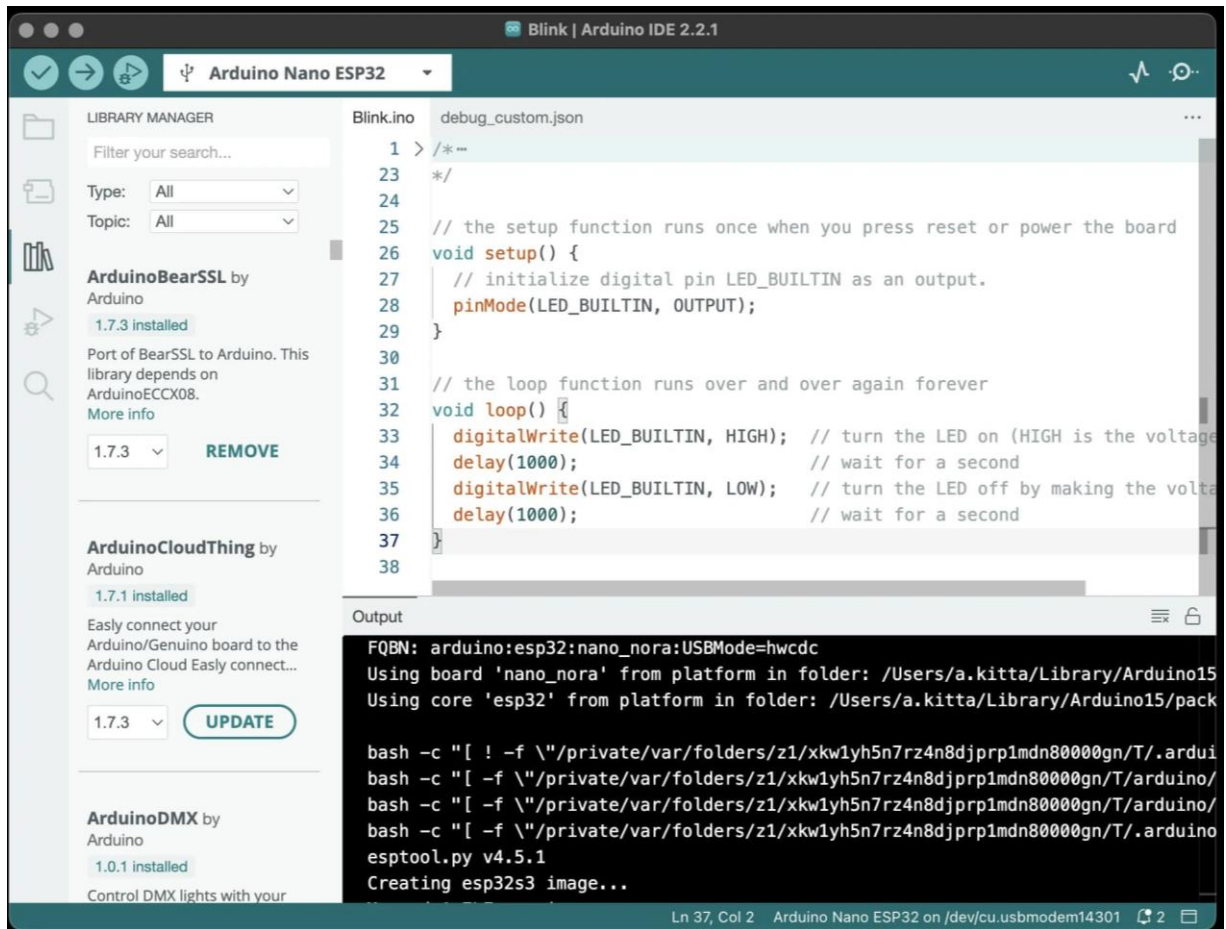


Рисунок 1.5 – Робоче вікно Arduino IDE [5]

Основні можливості Arduino IDE:

- редагування коду: простий текстовий редактор з підсвічуванням синтаксису; автоматичне форматування коду; вбудовані шаблони прикладів для швидкого старту [5];
- компіляція та завантаження: перетворення коду на машинний (байт-код) одним кліком; завантаження коду на плату через USB; вивід повідомлень про помилки та компіляцію [5];

- моніторинг даних: Serial Monitor — вікно для перегляду та надсилання серійних повідомлень між ПК та мікроконтролером; Serial Plotter — графічний інструмент для візуалізації даних у реальному часі [5];
- підтримка великої кількості плат: Arduino Uno, Mega, Nano, Leonardo, Due та багато інших; підтримка сторонніх плат (ESP8266, ESP32, STM32 тощо) через встановлення плат через менеджер плат [5];
- менеджер бібліотек: встановлення та оновлення тисяч бібліотек із репозиторію Arduino; легка інтеграція драйверів та зовнішніх пристроїв (дисплеї, датчики, двигуни тощо) [5];
- кросплатформеність: працює на Windows, macOS, Linux; існує як класична версія, так і Arduino IDE 2.0 з новими можливостями (автодоповнення, вкладки, теми, налагодження тощо) [5];
- підключення зовнішніх розширень: плагіни для роботи з дебагерами; підтримка роботи з PlatformIO через сторонні інструменти [5].

1.8 Wokwi

Wokwi – це безкоштовний онлайн-симулятор (рис. 1.6) для проектування, тестування та налагодження електронних схем на базі мікроконтролерів, таких як Arduino, ESP32, STM32 та Raspberry Pi Pico. Цей інструмент дозволяє створювати інтерактивні проекти без потреби у фізичному обладнанні, що робить його ідеальним для навчання, прототипування та віддаленої розробки [3].

Ключові особливості Wokwi:

- працює у браузері: не потребує встановлення — достатньо перейти на сайт <https://wokwi.com> [6];
- велика бібліотека компонентів: понад 100 віртуальних елементів, включно з сенсорами, дисплеями, сервомоторами, матрицями світлодіодів, кнопками тощо [6];

- реалістична симуляція: поведінка пристроїв і сигналів відтворюється близько до реальних умов [6];
- підтримка мов програмування: можна писати код на Arduino C/C++, MicroPython, або CircuitPython [6];
- інтерактивна симуляція: можна натискати кнопки, вводити значення, відстежувати сигнали тощо — все у реальному часі [6].

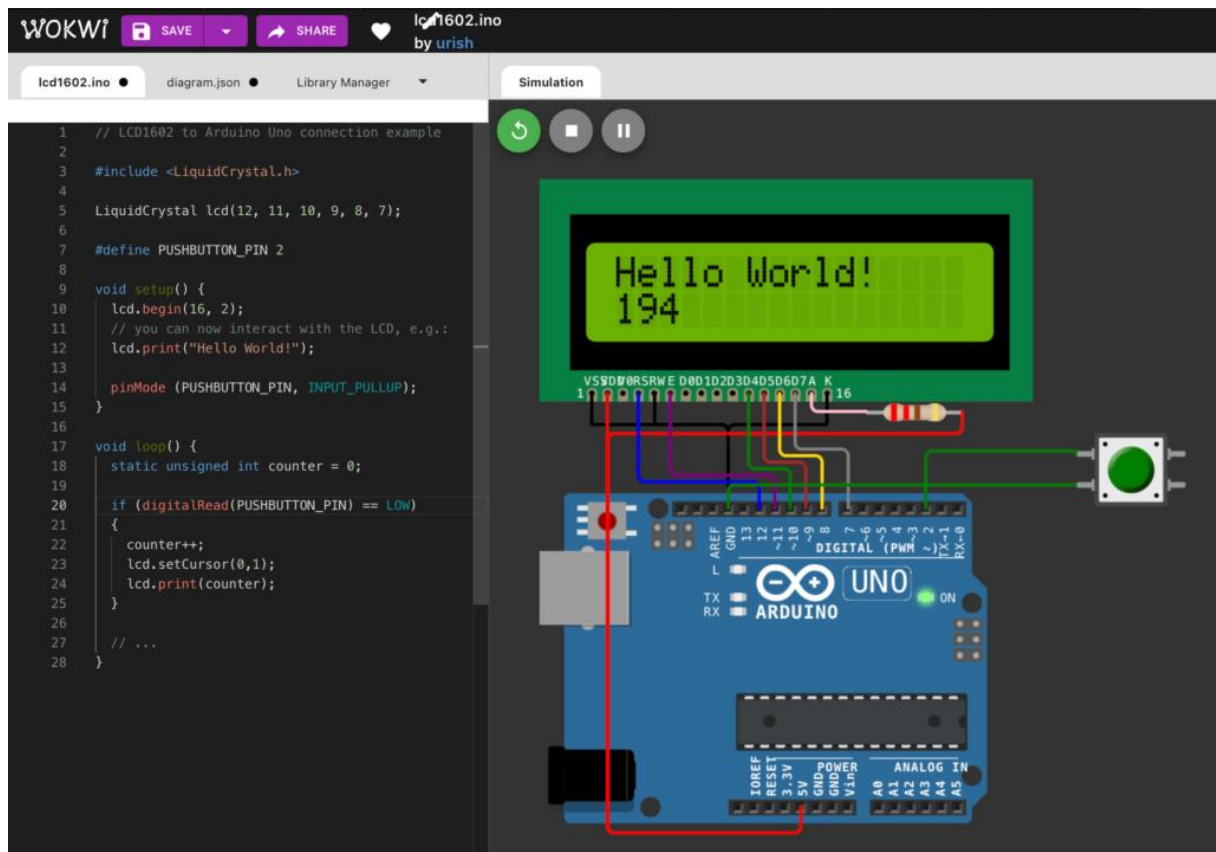


Рисунок 1.6 – Робоче вікно Wokwi [3]

Основні можливості Wokwi [6]:

- підтримка популярних платформ: Arduino Uno, Mega, Nano, ATtiny85, ESP32, Raspberry Pi Pico (у тому числі з MicroPython) та STM32;
- розширена бібліотека компонентів: дисплеї (LCD, OLED), сенсори (температури, відстані), кнопки, світлодіоди, серводвигуни, реле, матриці NeoPixel, модулі Wi-Fi та інші;

- інтерактивна симуляція: можливість взаємодії з елементами схеми в реальному часі — натискання кнопок, перемикання тумблерів, перегляд даних на дисплеях тощо;
- редактор коду з автодоповненням: підтримка Arduino, MicroPython, C/C++ (для Raspberry Pi Pico SDK), а також можливість використання зовнішніх бібліотек;
- віртуальний логічний аналізатор: для відстеження цифрових сигналів (UART, I2C, SPI) та їх аналізу без додаткового обладнання;
- інтеграція з Visual Studio Code: розширення дозволяє запускати симуляції безпосередньо з VS Code, підтримуючи такі фреймворки, як ESP-IDF, PlatformIO, Arduino CLI та інші;
- симуляція SD-карти – Зберігайте та витягуйте файли й каталоги з вашого коду. Платні користувачі також можуть завантажувати бінарні файли (наприклад, зображення);
- розширене налагодження за допомогою GDB – потужний налагоджувач Arduino та Raspberry Pi Pico для досвідчених користувачів;
- підтримка Wi-Fi симуляції: можливість тестування мережевих протоколів (MQTT, HTTP, NTP) у віртуальному середовищі.

2 РОЗРОБКА ЕЛЕКТРОННОГО ГОДИННИКА

2.1 Мета проєкту

Розробити та реалізувати прототип електронного годинника з використанням платформи Arduino в онлайн симуляторі Wokwi, який буде відображати поточний час та за потреби додаткові функції (наприклад, температура, календар тощо).

2.2 Компоненти

Для розробки прототипу електронного годинника необхідно:

- плата Arduino Uno;
- модуль годинника реального часу DS1307 RTC;
- матричний дисплей 32 x 8 пікселів;
- датчик температури та вологості DHT22.

Arduino Uno (рис. 2.1-2.2) – це популярна мікроконтролерна плата на базі мікроконтролера ATmega328P, яка широко використовується для навчання, прототипування та створення електронних пристроїв [7].

Основні технічні характеристики плати Arduino Uno [7]:

- мікроконтролер: ATmega328P;
- робоча напруга: 5 В;
- вхідна напруга (рекомендована): 7-12 В;
- цифрові входи/виходи: 14 (з них 6 можуть використовуватись як ШІМ-виходи);
- аналогові входи: 6;
- максимальний струм на пін I/O: 40 мА;
- пам'ять Flash: 32 КБ (з них 0.5 КБ використовується завантажувачем);
- оперативна пам'ять (SRAM): 2 КБ;

- EEPROM: 1 КБ;
- тактова частота: 16 МГц;
- порти зв'язку: USB, UART, I2C, SPI;
- інтерфейс програмування: USB через мікроконтролер ATmega16U2 (емуляція COM-порту).

Додатково:

- легко програмується через середовище Arduino IDE;
- має стандартний роз'єм живлення, USB-порт типу В;
- сумісна з різними модулями та шилдами (LCD, Wi-Fi, датчики тощо);
- ідеально підходить для створення проектів початкового та середнього рівня складності.



Рисунок 2.1 – Плата Arduino Uno [7]

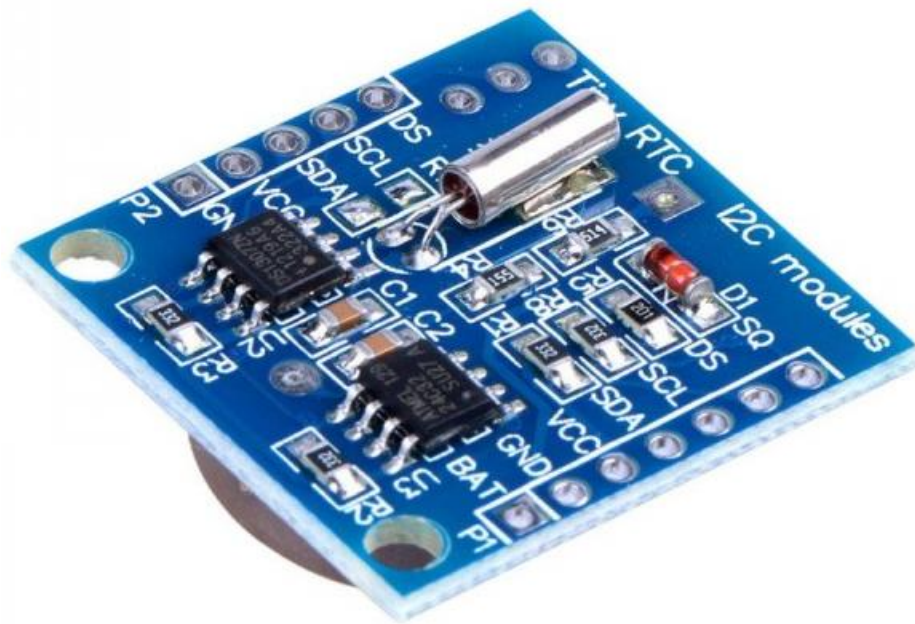


Рисунок 2.3 – DS1307 RTC [8]

Цифровий датчик температури та вологості підвищеної точності – датчик DHT22 (рис. 2.4) має заводське калібрування і характеризується низьким енергоспоживанням [9].



Рисунок 2.4 – DHT22 [9]

Характеристика датчика: виробник: ASAIR; тип: AM2302 цифровий; точність: 0,1 °C; діапазон вимірювання вологості: 0-100%; діапазон виміру температури: -40 ~ 80 °C; точність вимірювання вологості: ± 2% RH; точність

вимірювання температури: ± 0.5 градуса; напруга живлення: 3.6-6 В; кількість виводів: 4; ультранизьке енергоспоживання; не вимагає обв'язки; здатний працювати при досить довгому дроті [9].

Матричний LED-модуль на MAX7219 (рис. 2.5) – професійне рішення для створення динамічних світлодіодних дисплеїв та рухомих рядків. Модуль являє собою єдину друковану плату з чотирма послідовно з'єднаними матрицями 8x8 (загальна роздільна здатність 32x8 пікселів), керованими надійними драйверами MAX7219. Ідеально підходить для Arduino-проектів, інформаційних табло, годинників та рекламних дисплеїв. Підтримує каскадне підключення для створення дисплеїв необмеженої довжини. Модуль оснащений спеціальними насічками для можливості розділення на окремі матриці 8x8 при необхідності [10].

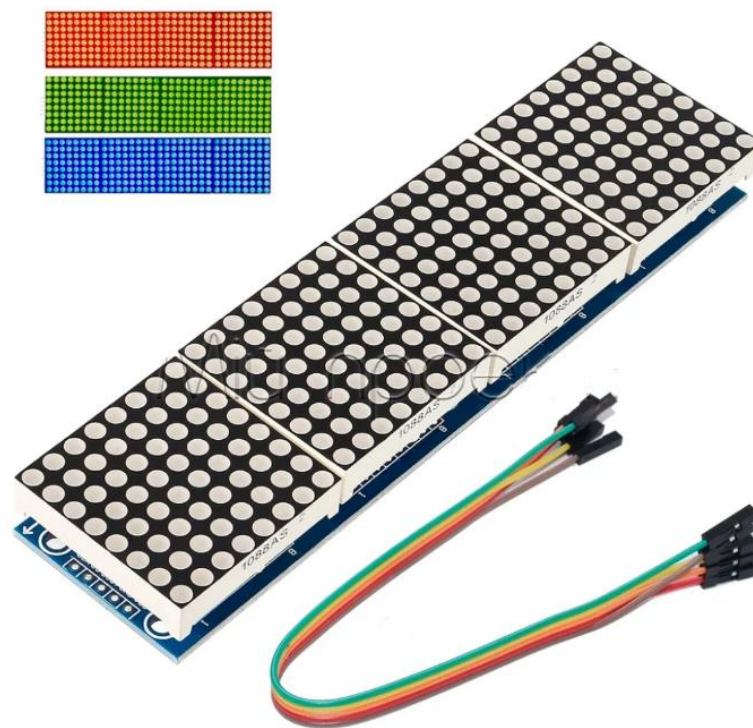


Рисунок 2.5 – LED-модуль на MAX7219 [10]

Детальні технічні характеристики: загальна роздільна здатність: 32x8 пікселів (світлодіодів); кількість матриць: 4 шт. (8x8 кожна); драйвер:

MAX7219 (по одному на кожен модуль); тип матриць: 1088AS червоні, зелені або сині світлодіоди; напруга живлення: 5В постійного струму; інтерфейс підключення: SPI (3-провідний); кількість пінів підключення: 5 (VCC, GND, DIN, CS, CLK); каскадне підключення (піни IN: для підключення до контролера; піни OUT: для підключення наступного модуля); модульність: можливість розділення на 4 окремі модулі 8x8; сумісність: Arduino, Raspberry Pi, інші мікроконтролери; підтримувані бібліотеки (Adafruit_GFX + Max72xxPanel (англійський текст); LedControl (кириличний текст); SPI.h (низькорівневе управління)); програмування: Arduino IDE, PlatformIO; колір світлодіодів: червоний, зелений або синій; конструкція: єдина друкована плата з насічками для розділення; рекомендована швидкість прокрутки: 75 мс (регулюється програмно) [10].

Технічні переваги модуля:

- каскадне підключення – спеціальні піни IN/OUT дозволяють легко з'єднувати декілька модулів для створення довгих дисплеїв без додаткового обладнання [10];

- надійні драйвери MAX7219 – кожна матриця керується окремим чіпом MAX7219, що забезпечує стабільну роботу та високу яскравість світлодіодів [10];

- модульна конструкція – наявність насічок дозволяє розділити плату на чотири окремі модулі 8x8 для гнучкого використання в різних проєктах [10];

- підтримка кирилических символів – можливість створення власних шрифтів для відображення українського, російського та інших кирилических текстів [10];

- простота підключення – всього 5 пінів для підключення до Arduino та інших мікроконтролерів через SPI-інтерфейс [10];

- готові бібліотеки – наявність множини готових бібліотек для Arduino IDE, включаючи Adafruit_GFX та LedControl для швидкого старту проєктів [10].

Широкі можливості застосування.

Інформаційні системи – створення динамічних табло для відображення новин, курсів валют, погоди або розкладів транспорту. Модуль дозволяє легко програмувати швидкість прокрутки та ефекти анімації. Підтримує багаторядковий текст та графічні елементи для максимальної інформативності дисплею [10].

Рекламні та промоційні дисплеї – ідеальний вибір для створення яскравих рекламних стрічок у вітринах магазинів, кафе або офісах. Можливість каскадного підключення дозволяє створювати дисплеї будь-якої довжини. Червоні світлодіоди забезпечують високу видимість навіть при яскравому освітленні [10].

Освітні проєкти та прототипування – відмінний інструмент для вивчення основ програмування мікроконтролерів, роботи з SPI-інтерфейсом та створення інтерактивних проєктів. Наявність готових бібліотек дозволяє швидко почати експерименти, а можливість створення власних шрифтів розвиває навички низькорівневого програмування [10].

Декоративні та мистецькі інсталяції – використання для створення світлових композицій, інтерактивних арт-об'єктів або декоративних елементів інтер'єру. Модуль може відображати не тільки текст, але й графічні патерни, анімації та реагувати на зовнішні датчики для створення інтерактивних інсталяцій [10].

Промислові індикатори та контролери – застосування в системах моніторингу та контролю для відображення статусу обладнання, параметрів процесів або попереджень. Надійність драйверів MAX7219 забезпечує стабільну роботу в промислових умовах, а можливість програмування різних режимів відображення підвищує функціональність системи [10].

Важливі аспекти використання.

Підключення живлення – обов'язково використовуйте стабілізоване живлення 5В. При підключенні декількох модулів каскадом рекомендується використовувати зовнішній блок живлення для забезпечення достатнього

струму. Максимальний струм споживання одного модуля може досягати 200-300 мА при повному засвічуванні всіх світлодіодів [10].

Програмування кириличних символів – для відображення українського та російського тексту необхідно використовувати бібліотеку LedControl та створювати власні шрифти. Кожен символ описується масивом байтів, де кожен біт відповідає одному світлодіоду. Рекомендується спочатку створити таблицю відповідності англійських і кириличних літер [10].

Каскадне підключення – при з'єднанні декількох модулів важливо правильно підключити піни IN/OUT. Сигнал від мікроконтролера подається на піни IN першого модуля, а піни OUT з'єднуються з пінами IN наступного модуля. У програмному коді необхідно вказати загальну кількість підключених модулів для правильної ініціалізації [10].

Розділення модуля – якщо потрібно розділити модуль на окремі матриці 8x8, використовуйте спеціальні насічки на платі. Розрізайте акуратно, використовуючи канцелярський ніж або спеціальний інструмент. Після розділення кожна матриця буде працювати незалежно і потребуватиме окремого підключення [10].

Налаштування яскравості – драйвери MAX7219 підтримують програмне регулювання яскравості світлодіодів. Використовуйте функцію `setIntensity()` для встановлення оптимального рівня яскравості залежно від умов освітлення. Зниження яскравості також дозволяє зменшити споживання електроенергії [10].

2.3 Симуляція

На рис. 2.6 представлено інтерфейс симулятора Wokwi.

Вікно симулятора (праворуч) [11]:

- діаграма (Diagram Editor) – дозволяє вручну додавати та з'єднувати компоненти натисканням на «+» або за допомогою JSON-редагування;

- панель запуску – зелена кнопка “Play” запускає компіляцію й симуляцію; “Stop” зупиняє її;

– інтерактивність – можна вмикати/вимикати кнопки, перемикачі, моніторити поведінку, як у реальному середовищі.

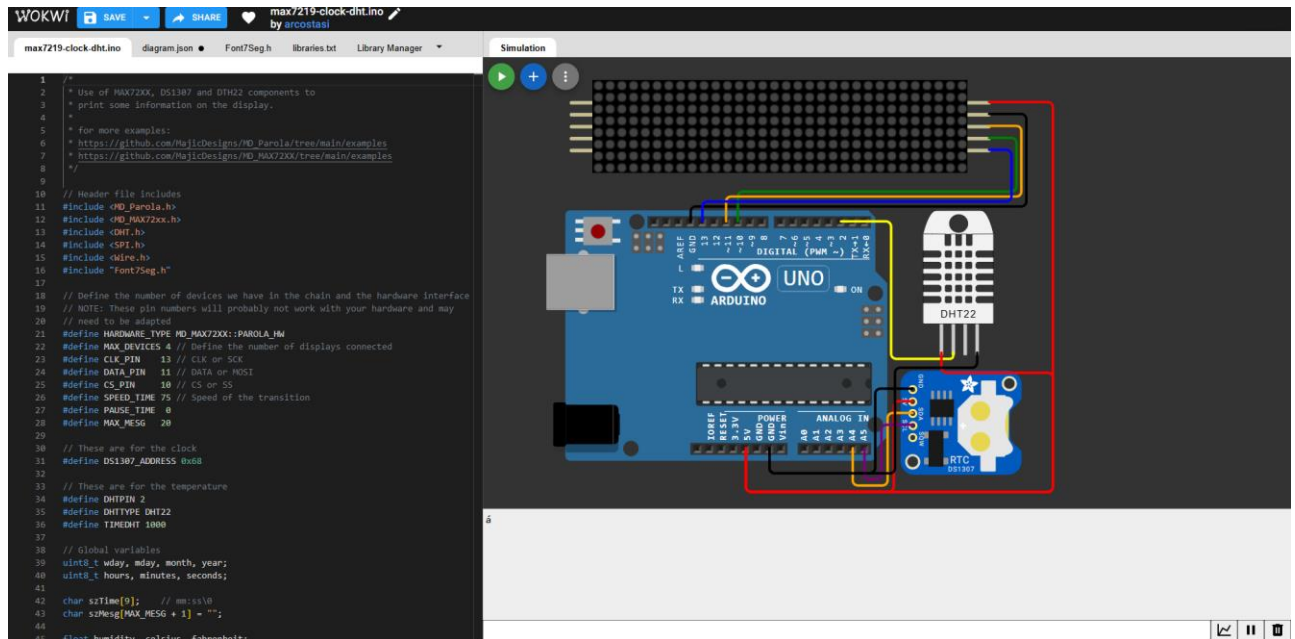


Рисунок 2.6 – Інтерфейс симулятора Wokwi

Редактор коду (зліва) [11]:

– редагування sketch — тут пишеш Arduino-код (.ino), можна створювати вкладки;

– автодоповнення — підказки для класів, функцій, бібліотек.

Менеджер бібліотек [11]:

– дозволяє додавати бібліотеки через вкладку Library Manager або завантажити ZIP-файли.

Редагування JSON (diagram.json) [11]:

– за допомогою вкладки diagram.json можна деталізовано налаштовувати компоненти – змінювати пінування, типи, атрибути (наприклад, I2C vs Full LCD).

Додаткові інструменти [11]:

– Virtual Logic Analyzer – аналіз цифрових сигналів (UART, I2C, SPI)

- Serial Monitor – виводить дані з Serial.print(), підтримує зовнішні серійні порти;
- підтримка Wi-Fi, SD-карт, GDB-налагодження для ESP32/STM32 через VS Code.

На рис. 2.7 представлено в загальному вигляді в якій послідовності будуть показані значення на матричному дисплеї годинника.

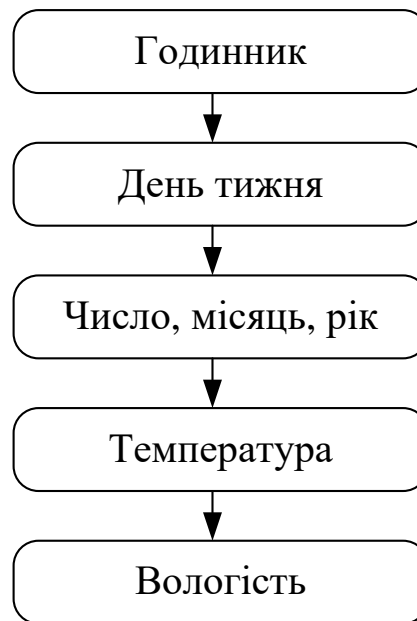


Рисунок 2.7 – Послідовність значень

На рис. 2.8-2.13 представлено відображення всіх функціональних значень, які з’являються рухомим рядком у послідовності відповідно рис. 2.7.

З рис. 2.8-2.13 бачимо, що розроблений годинник з додатковими функціями працює, але постає питання чи правильні значення він надає, тобто чи демонструє застосування підтягування інформації за допомогою мережі інтернет (Wi-Fi).

Продемонструємо повноцінну працездатність роботи електронного годинника (рис. 2.14-2.17).

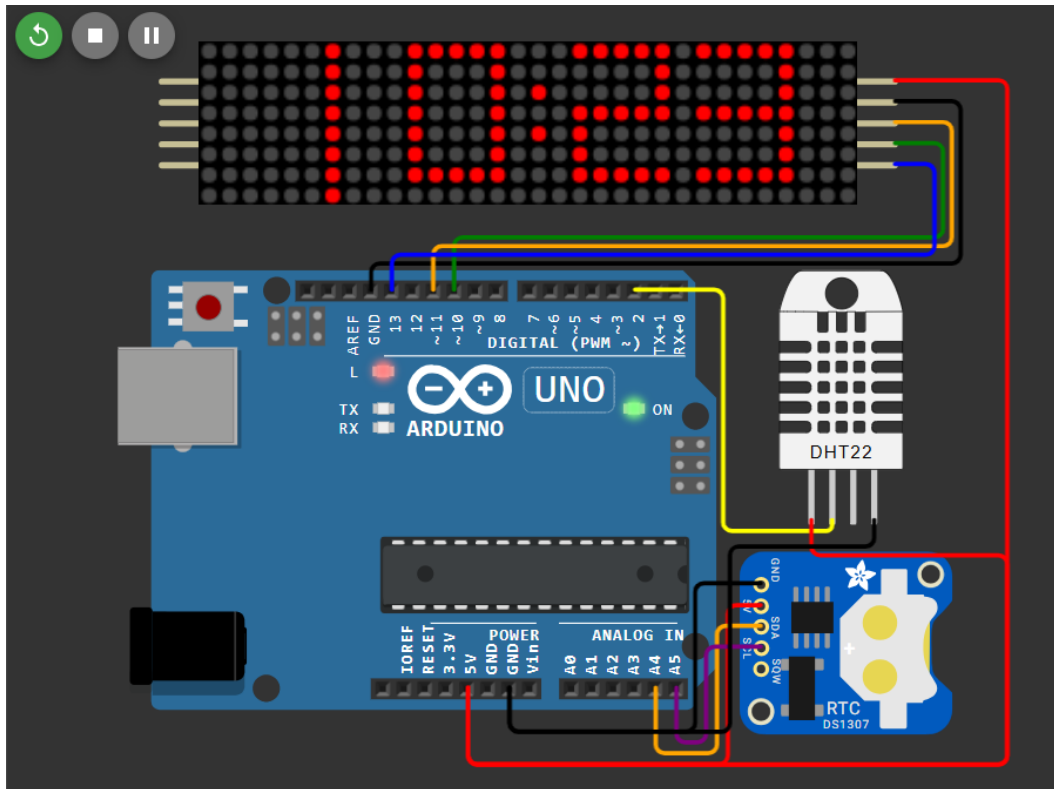


Рисунок 2.8 – Годинник

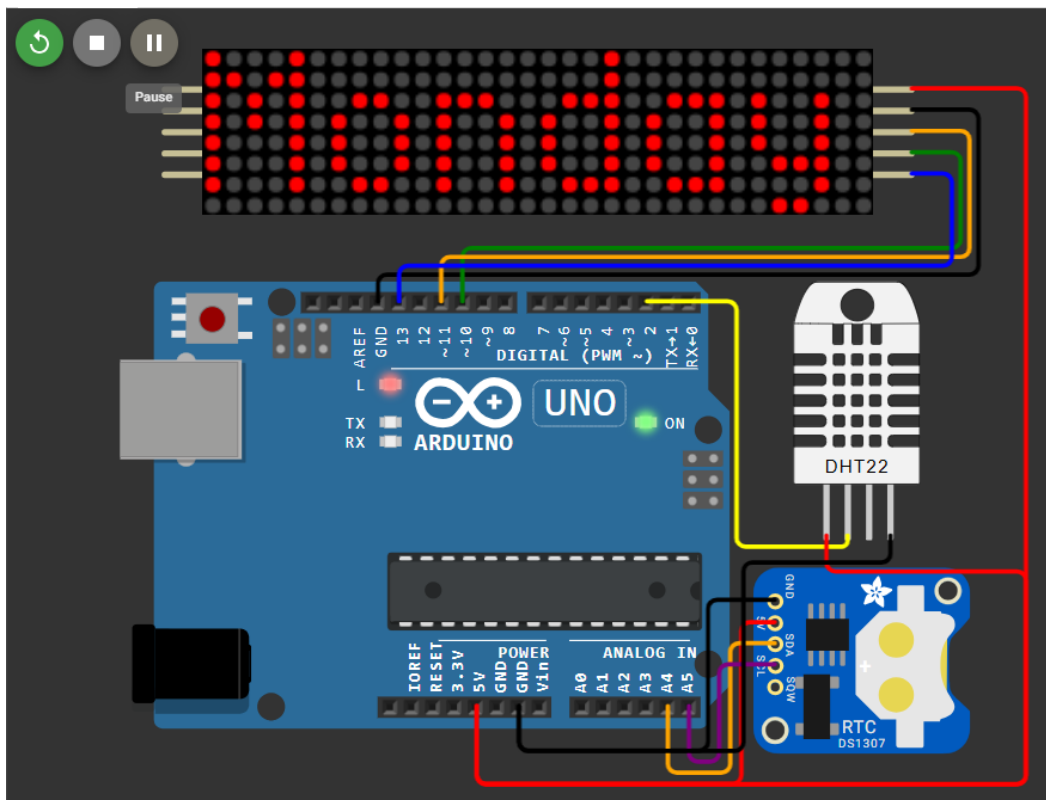


Рисунок 2.9 – День тижня

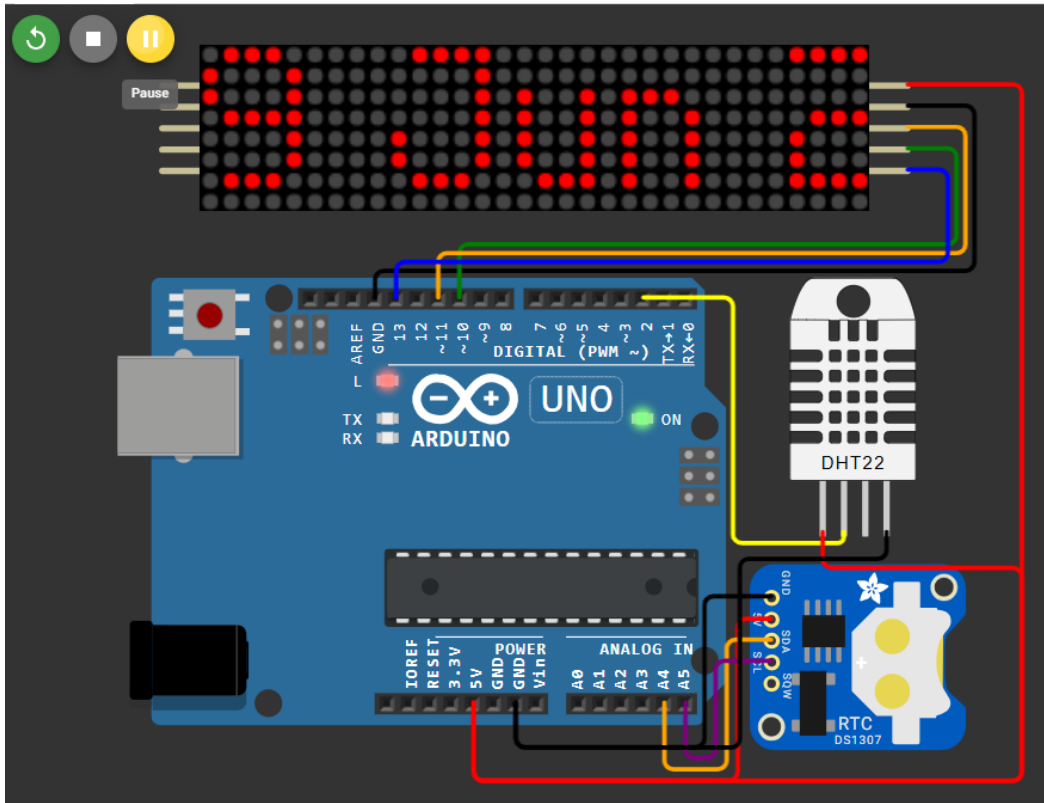


Рисунок 2.10 – Число, місяць, рік

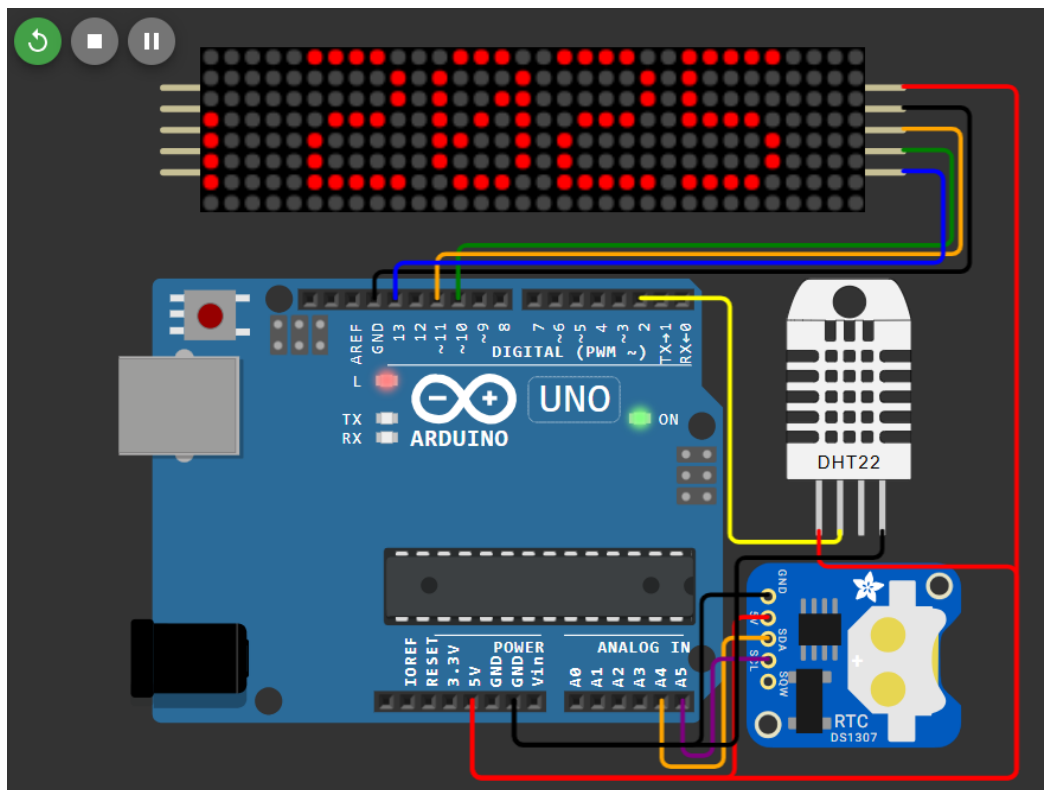


Рисунок 2.11 – Число, місяць, рік

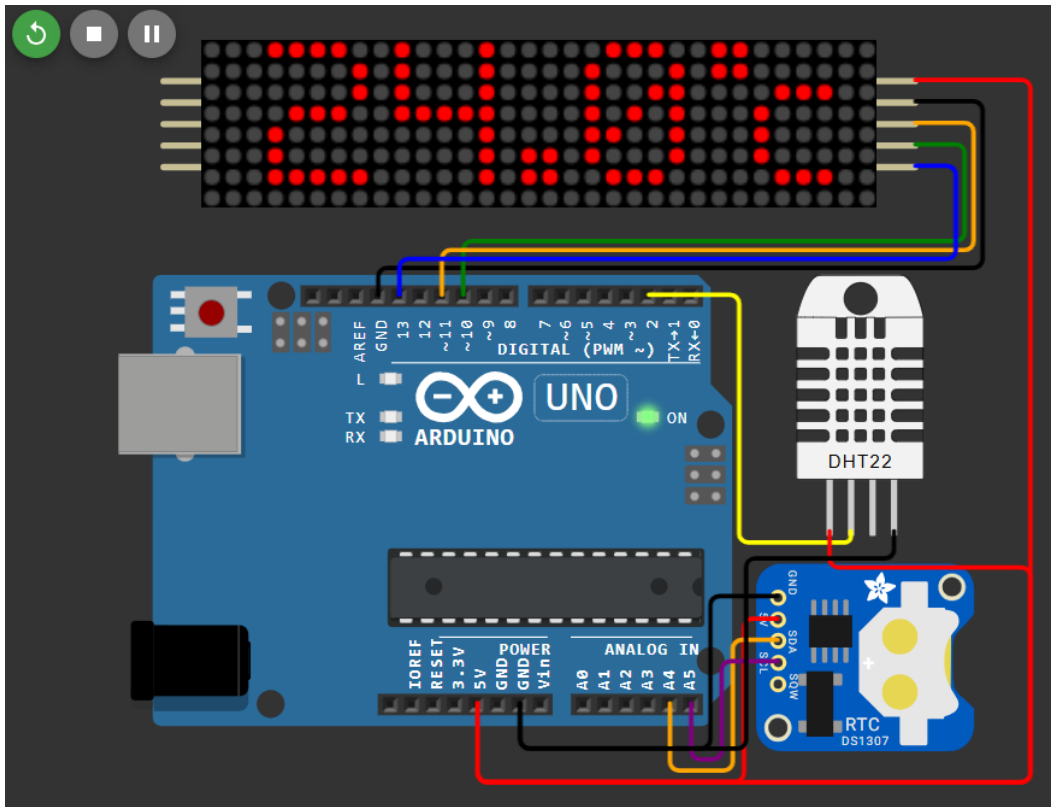


Рисунок 2.12 – Температура

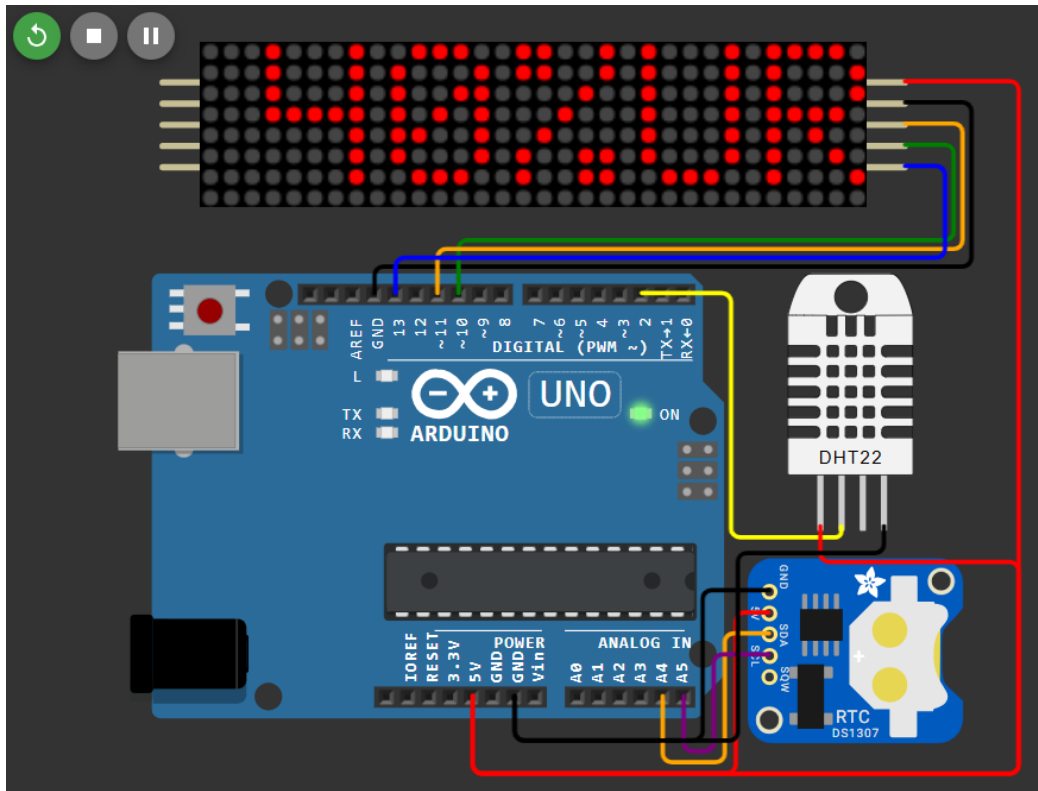


Рисунок 2.13 – Вологість

На рис. 2.14-2.17 представлено підтвердження працездатності, для цього під час симуляції роботи годинника було відкрито вбудований календар на комп'ютері.

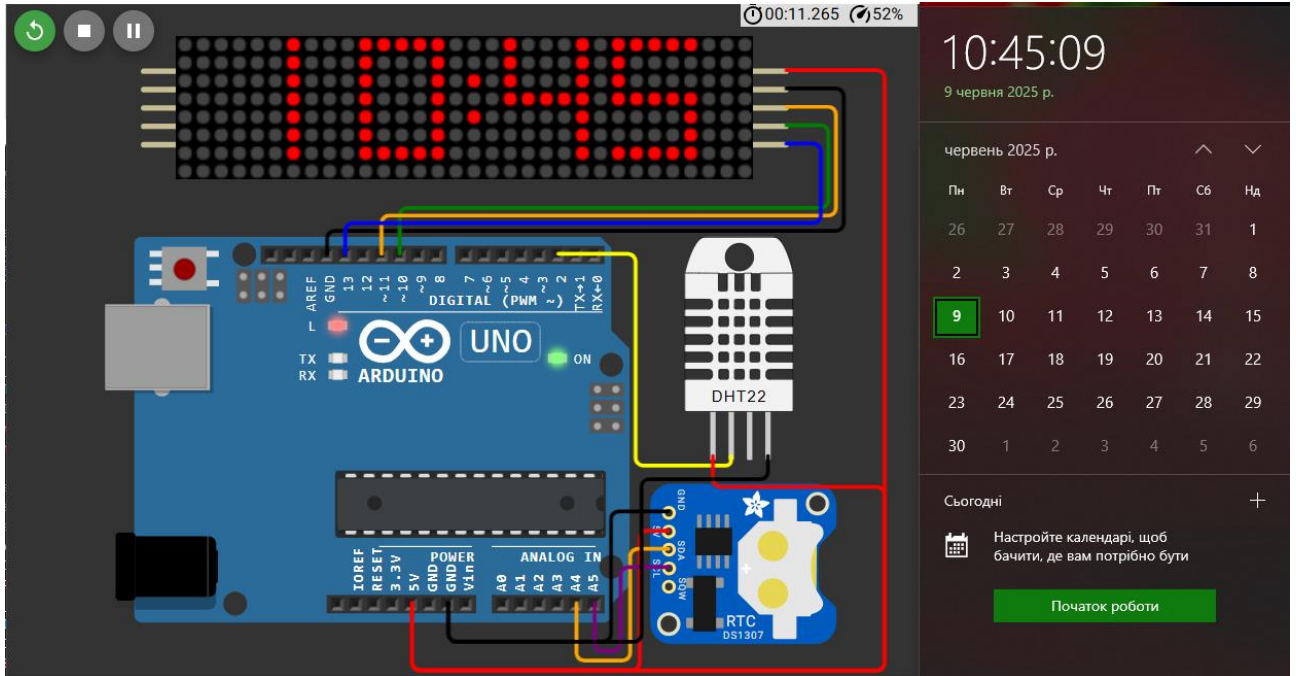


Рисунок 2.14 – Годинник

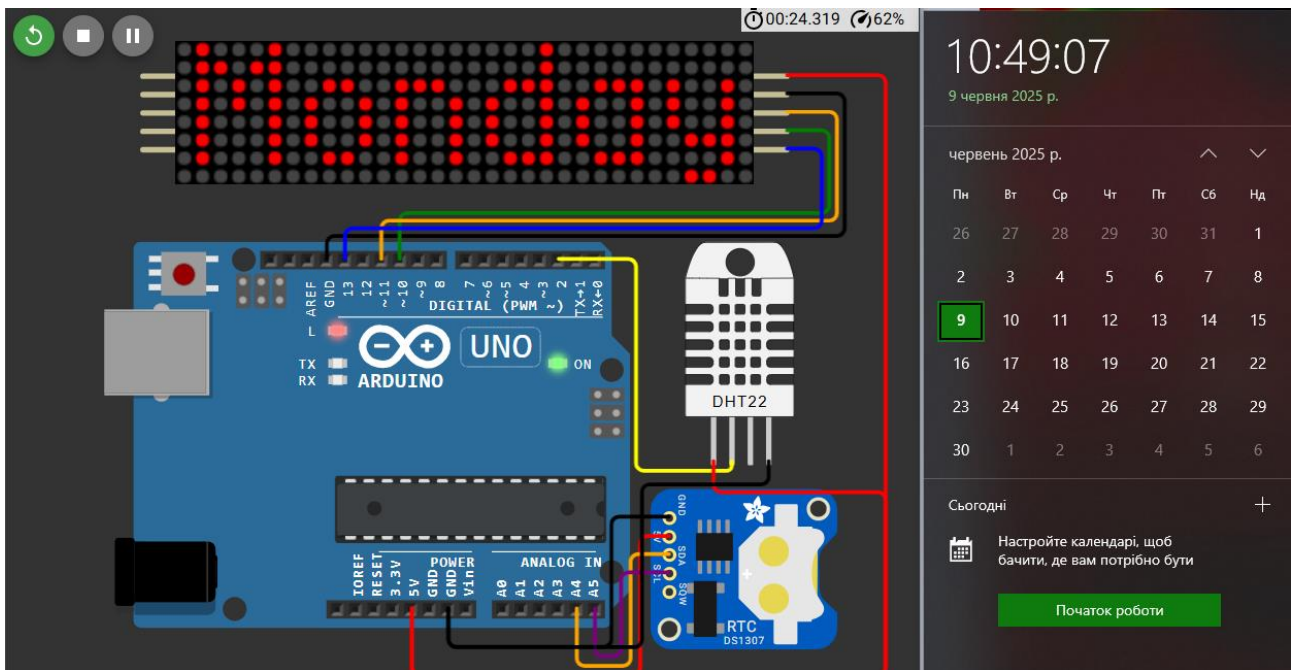


Рисунок 2.15 – День тижня

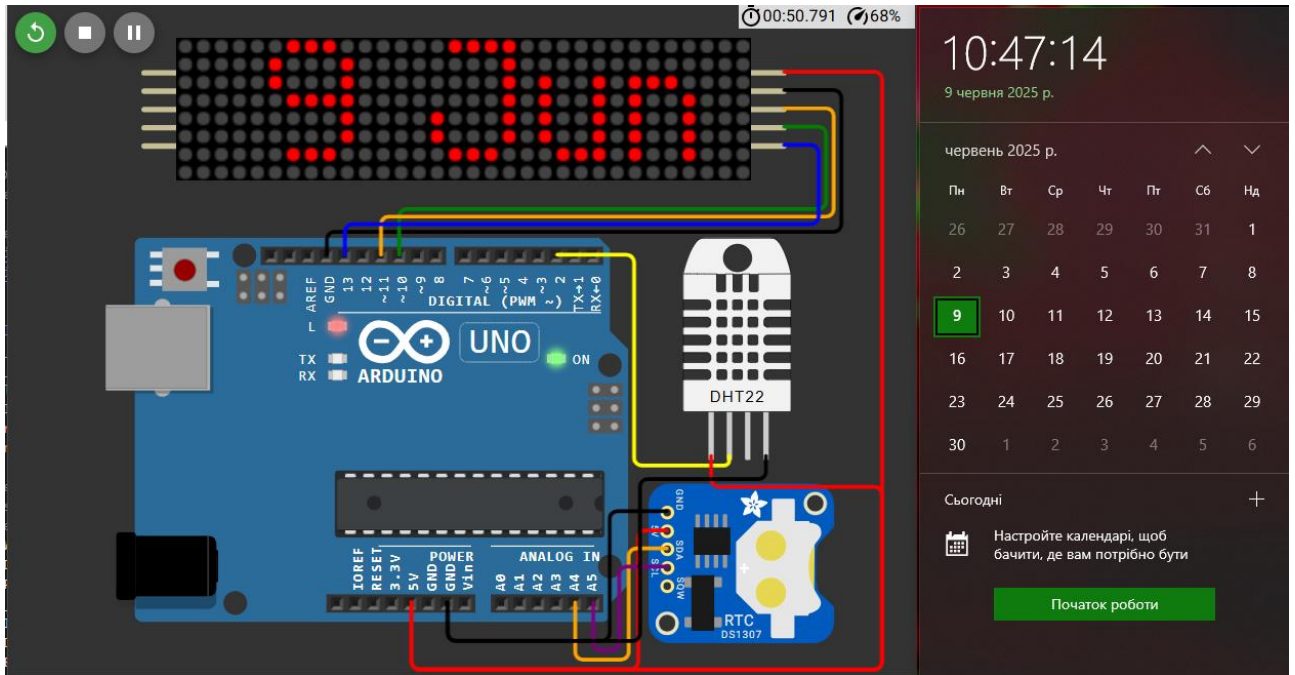


Рисунок 2.16 – Дата та місяць

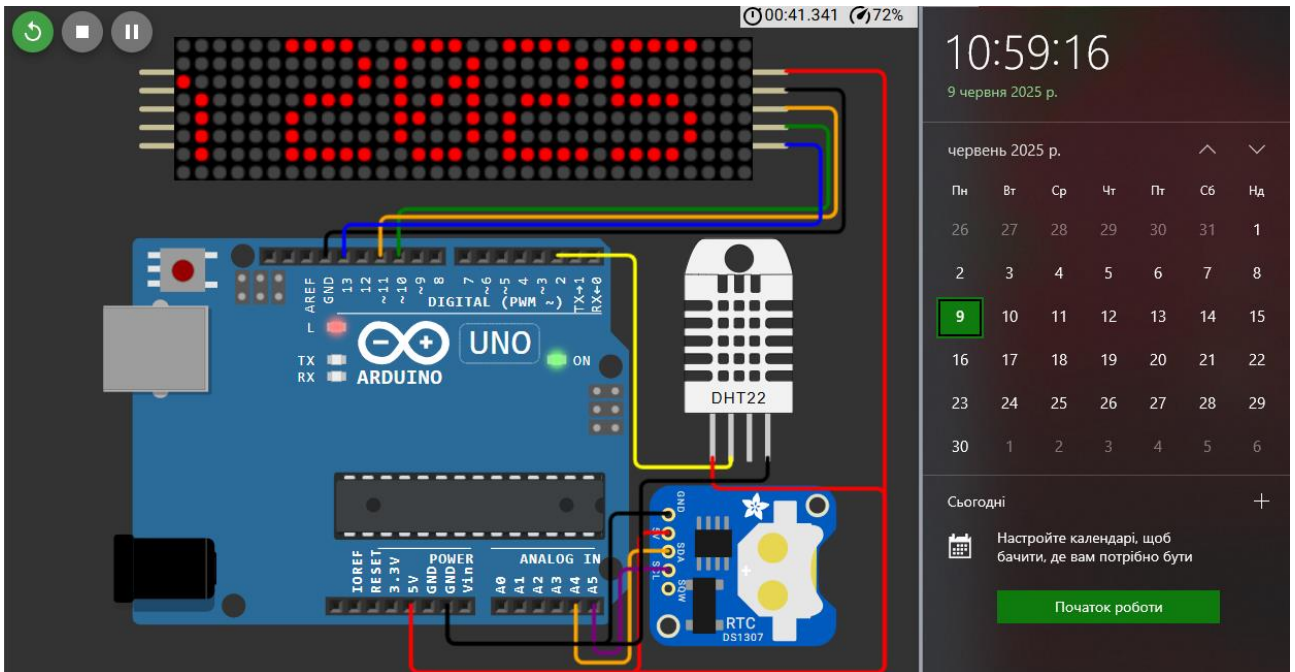


Рисунок 2.17 – Рік

Для підтвердження таких показників як температура та вологість, необхідно робити фізичний прототип електронного годинника, щоб підтвердити правильність функціонування і цих показників.

Годинник можна модернізувати і додати також датчики тиску, таким чином електронний годинник виконуватиме роль домашньої метеостанції. Також можна змінити тип дисплею та відображення даних, все це індивідуально.

Електронний годинник на базі Arduino має широкий спектр застосувань – як у побуті, так і в навчанні чи спеціалізованих проєктах.

Дану розробку можна впровадити в загальний проєкт «Розумного будинку», такий електронний годинник буде гарним доповненням.

«Розумний будинок» (Smart Home) – це житлове приміщення, оснащене системами автоматизації, що керуються через смартфон, комп'ютер або голосові команди. Такі системи дозволяють управляти освітленням, опаленням, безпекою, побутовою технікою та іншими пристроями дистанційно або автоматично.

2.4 Живлення

Живлення від USB (5V через USB-порт) – джерелом живлення може виступати комп'ютер, павербанк, адаптер USB.

Переваги:

- зручно під час розробки;
- завжди стабільні 5 В.

Недоліки:

- потрібно постійне підключення до джерела USB;
- неавтономне рішення.

Живлення від батарей/акумуляторів:

– Li-ion/Li-Po акумулятор (3.7 В) – через плату підвищення напруги (Boost Converter) або спеціальні модулі типу TP4056 + Step-Up (переваги: компактність, перезаряджуваність; недоліки: потрібно контролювати заряд/розряд, додаткові плати);

– батарейки типу AA/AAA (4.5–6 В) – 3-4 батарейки під'єднані послідовно (переваги: легко замінити, автономність; недоліки: великі габарити, низька ємність).

Адаптер змінного струму (блок живлення 9В-12В):

– через роз'єм живлення DC jack (барельний роз'єм на платі Arduino Uno).

– Arduino має вбудований стабілізатор, який перетворює 9-12 В у 5 В.

Переваги: стабільність, надійність для стаціонарних рішень. Недоліки: залежність від електромережі, не підходить для мобільних пристроїв.

Кнопкова батарейка (для RTC-модуля) – зазвичай використовується CR2032 для модуля годинника реального часу (RTC, напр. DS3231).

Функція: підтримує хід часу, коли живлення Arduino вимкнене.

Переваги: забезпечує безперервність часу навіть при вимкненні пристрою. Недоліки: не живить саму плату Arduino – лише модуль RTC.

Альтернативні варіанти:

– сонячна панель + акумулятор – для автономних outdoor-проектів;

– павербанк – зручно як автономне джерело 5 В для короткочасної роботи.

В табл. 2.1 представлені можливі варіанти живлення для електронного годинника.

Таблиця 2.1 – Варіанти живлення для електронного годинника

Джерело живлення	Напруга	Автономність	Застосування
USB	5 В	<input type="checkbox"/>	Розробка, тестування
Адаптер 9–12 В	9–12 В	<input type="checkbox"/>	Стаціонарний годинник
AA/AAA батарейки	4.5–6 В	<input type="checkbox"/>	Простий автономний проект
Li-ion / Li-Po	3.7 В (Boost)	<input type="checkbox"/>	Компактні автономні рішення
CR2032 (для RTC)	~3 В	<input type="checkbox"/> (для часу)	Збереження ходу часу
Сонячна панель + акумулятор	Залежить	<input type="checkbox"/>	Зовнішні пристрої

В табл. 2.2 представлено порівняльний аналіз електронного годинника на базі Arduino та звичайного електронного годинника.

Таблиця 2.2 – Порівняльний аналіз

Критерій	Arduino-годинник	Простий елект. годинник
Гнучкість / Модульність	<input type="checkbox"/> Повністю налаштований (можна додавати функції: дата, температура, Wi-Fi, будильник, тощо)	<input type="checkbox"/> Обмежений базовим функціоналом
Навчальна цінність	<input type="checkbox"/> Ідеально для вивчення електроніки, програмування, логіки	<input type="checkbox"/> Не навчає, готовий продукт
Вартість	<input type="checkbox"/> Часто дорожчий (Arduino + модулі + дисплей)	<input type="checkbox"/> Дешевий (1–5 доларів у масовому виробництві)
Енергоефективність	<input type="checkbox"/> Потребує більше енергії, особливо з LCD, датчиками тощо	<input type="checkbox"/> Дуже економний, часто працює роками від однієї батарейки
Автономність	<input type="checkbox"/> Залежить від акумуляторів чи зовнішнього живлення	<input type="checkbox"/> Працює від компактної батарейки
Простота у використанні	<input type="checkbox"/> Потрібно знання електроніки, коду, збирання схеми	<input type="checkbox"/> Просто: увімкнув — і працює
Розширення функціоналу	<input type="checkbox"/> Можна додати Bluetooth, Wi-Fi, GPS, NTP-синхронізацію тощо	<input type="checkbox"/> Неможливо змінити чи доповнити
Розміри	<input type="checkbox"/> Зазвичай більший, громіздкий (особливо прототипи)	<input type="checkbox"/> Компактний, портативний
Надійність	<input type="checkbox"/> Потребує ручного складання, може бути менш стабільним	<input type="checkbox"/> Перевірене заводське виробництво
Гнучкість / Модульність	<input type="checkbox"/> Легко ремонтувати, замінити модуль або перепрошити	<input type="checkbox"/> Неремонтопридатний або складний в ремонті
Сервісність	<input type="checkbox"/> Повністю налаштований (можна додавати функції: дата, температура, Wi-Fi, будильник, тощо)	<input type="checkbox"/> Обмежений базовим функціоналом

ВИСНОВКИ

Онлайн та офлайн симулятори Arduino значно спрощують розробку, тестування та навчання роботі з мікроконтролерами. Онлайн-симулятори (як-от Wokwi) є надзвичайно зручними завдяки доступності з будь-якого пристрою без встановлення програмного забезпечення, а також швидкому обміну проектами через інтернет. Вони ідеальні для навчання, експериментів та швидкого прототипування.

Натомість офлайн-симулятори (наприклад, Proteus або Tinkercad Desktop) часто пропонують більше функцій, точнішу симуляцію аналогових процесів, глибший контроль над середовищем та підтримку складних електронних схем. Вони краще підходять для професійної розробки, коли потрібна висока точність та незалежність від інтернет-з'єднання.

Отже, вибір між онлайн і офлайн-симуляторами залежить від цілей користувача: для навчання та швидкого старту зручно користуватись онлайн-середовищами, тоді як для складних інженерних задач доцільніше обирати офлайн-інструменти. У багатьох випадках поєднання обох підходів дає найкращий результат.

Для розробки електронного годинника на базі плати Arduino було використано онлайн симулятор Wokwi. Симуляція електронного годинника є працездатною та готова до фізичної реалізації.

Годинник можна модернізувати і додати також датчики тиску, таким чином електронний годинник виконуватиме роль домашньої метеостанції. Також можна змінити тип дисплею та відображення даних для індивідуальних вподобань.

Дану розробку можна впровадити в загальний проєкт «Розумного будинку», такий електронний годинник буде гарним його доповненням.

ПЕРЕЛІК ПОСИЛАНЬ

1. Використовуємо Arduino симулятор замість самої плати. [Електронний ресурс] – Режим доступу: <https://naurok.com.ua/metodichni-rekomendaci-vikoristovuemo-arduino-simulyator-zamist-samo-plati-78684.html>
2. Proteus. [Електронний ресурс] – Режим доступу: <https://arduinofactory.com/proteus-arduino-simulator/>
3. Онлайн та офлайн симулятори Arduino. [Електронний ресурс] – Режим доступу: <https://get3d.tech/articles/onlajn-ta-oflajn-symulyatory-arduino/>
4. Arduino IO Simulator Drag, Drop & Draw. [Електронний ресурс] – Режим доступу: <https://hobbytech.com.ua/arduino-io-simulator-drag-drop-draw/>
5. Огляд Arduino IDE. [Електронний ресурс] – Режим доступу: <https://kolotushkin.com/article.php?id=1>
6. Welcome to Wokwi! [Електронний ресурс] – Режим доступу: <https://docs.wokwi.com/>
7. Arduino UNO - популярна плата розробки. [Електронний ресурс] – Режим доступу: <https://itmaster.biz.ua/directory/kits-nabory/arduino-uno.html>
8. Модуль годинника реального часу DS1307 RTC. [Електронний ресурс] – Режим доступу: <https://ardushop.in.ua/arduino/real-time-clock-module-ds1307-rtc>
9. Датчик вологості та температури DHT22. [Електронний ресурс] – Режим доступу: <https://arduino.ua/prod301-datchik-vlajnosti-i-temperatyri-dht22?srsId=AfmBOoqzYCRL-NyMXHVtStpBtUDKDFxH5E6U2loCNKUCjCo7dazdsRR->
10. Матричний модуль MAX7219 для Arduino (32x8). [Електронний ресурс] – Режим доступу: https://myproject.com.ua/matrichnij-modul-max7219-BwE&utm_source=google&utm_medium=src&utm_campaign=Fiksovana_Stavka
11. Interactive Diagram Editor. [Електронний ресурс] – Режим доступу: https://docs.wokwi.com/guides/diagram-editor?utm_source