

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
АВТОМАТИЗОВАНИХ СИСТЕМ**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Проектування та реалізація системи розумного будинку на
основі Arduino»

на здобуття освітнього ступеня магістра
зі спеціальності 126 Інформаційні системи та технології
(код, найменування спеціальності)

освітньо-професійної програми 126 Інформаційні системи та технології
(назва)

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

_____ Ігор Білецький
(підпис) *Ім'я, ПРІЗВИЩЕ* здобувача

Виконав: здобувач вищої освіти гр. ІСД-62
Білецький Ігор
Ім'я, ПРІЗВИЩЕ

Керівник: Власенко В.О. Доцент каф. ПАУД
науковий ступінь, Ім'я, ПРІЗВИЩЕ
вчене звання

Рецензент: _____
науковий ступінь, Ім'я, ПРІЗВИЩЕ
вчене звання

Київ 2024

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
Навчально-науковий інститут інформаційних технологій**

Кафедра Інженерії програмного забезпечення автоматизованих систем
Ступінь вищої освіти Магістр
Спеціальність 126 Інформаційні системи та технології
Освітньо-професійна програма 126 Інформаційні системи та технології

ЗАТВЕРДЖУЮ
Завідувач кафедру Сторчак К.П.
_____ Ім'я, ПРІЗВИЩЕ
«___» _____ 20__ р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Білецький Ігор Сергійович
(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Проектування та реалізація системи розумного будинку на основі Arduino

керівник кваліфікаційної роботи Власенко В.О. Доцент каф. ПАУД,
(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «___» _____ 20__ р. №___

2. Строк подання кваліфікаційної роботи «___» _____ 20__ р.

3. Вихідні дані до кваліфікаційної роботи:

Наукова-технічна література по IoT та екосистемі arduino;

Нормативні матеріали

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Огляд Дротових протоколів комунікації;

2. Протоколи та інструменти обміну IoT даними;

3. Програмування мікроконтролерів arduino за допомогою Arduino IDE

5. Перелік ілюстративного матеріалу: *презентація*

6. Дата видачі завдання «___» _____ 20__ р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Огляд IoT та екосистеми arduino	02.10.2024	Виконано
2	Протоколи та інструменти обміну IoT даними	06.10.2024	Виконано
3	Детальний огляд сімейства мікроконтролерів	16.10.2024	Виконано
4	Використання дротових протоколів комунікації	28.10.2024	Виконано
5	Рекомендації щодо створення дротового розумного будинку на основі PLC	03.11.2024	Виконано
6	Управління енергією за допомогою Opta	12.11.2024	Виконано
7	Програмування мікроконтролерів arduino за допомогою Arduino IDE	01.12.2024	Виконано
8	Вступ, висновок	15.12.2024	Виконано

Здобувач(ка) вищої освіти

_____ (підпис)

_____ (Ім'я, ПРІЗВИЩЕ)

Керівник кваліфікаційної роботи

_____ (підпис)

_____ (Ім'я, ПРІЗВИЩЕ)

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут інформаційних технологій

**ПОДАННЯ
ГОЛОВІ ЕКЗАМЕНАЦІЙНОЇ КОМІСІЇ
ЩОДО ЗАХИСТУ КВАЛІФІКАЦІЙНОЇ РОБОТИ
на здобуття освітнього ступеня магістра**

Направляється здобувач Білецький І.С. до захисту кваліфікаційної роботи
(*прізвище та ініціали*)

за спеціальністю 126 Інформаційні системи та технології
(*код, найменування спеціальності*)

освітньо-професійної програми 126 Інформаційні системи та технології
(*назва*)

на тему: «Проектування та реалізація системи розумного будинку на
основі Arduino». Кваліфікаційна робота і рецензія додаються.

Директор ННІТ _____ Бондарчук А.П.
(*підпис*) (*Ім'я, ПРІЗВИЩЕ*)

Висновок керівника кваліфікаційної роботи

Здобувач(ка) дослідив та описав використання програмованих логічних
контролерів для проектування та реалізацію дротового розумного
будинку. Розглянув та описав сучасну серію PLC Arduino Opta, та
порівняв з альтернативними рішеннями на основі Arduino. Надав
рекомендації щодо проектування енергомережі та дротового розумного
будинку.

Все це дозволяє оцінити виконану кваліфікаційну роботу здобувача
Білецького Ігоря Сергійовича на оцінку «відмінно» та присвоїти йому
кваліфікацію _____.

Керівник кваліфікаційної роботи _____ Власенко В.О.
(*підпис*) (*Ім'я, ПРІЗВИЩЕ*)

«_____» _____ 20__ року

Висновок кафедри про кваліфікаційну роботу

Кваліфікаційна робота розглянута. Здобувач Білецький І.С. допускається
до захисту даної роботи в Екзаменаційній комісії.

Завідувач кафедру ІІЗАС _____ Сторчак К.П.
(*назва*) (*підпис*) (*Ім'я, ПРІЗВИЩЕ*)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: стор., рис., табл., джерел.

Мета роботи – проектування та реалізація системи розумного будинку на основі Arduino з метою створення надійного дротового розумного будинку.

Об'єкт дослідження – система розумного будинку на основі програмного логічного контролера Arduino Opta.

Предмет дослідження – аспекти створення дротового розумного будинку, враховуючи особливості та можливості роботи Arduino Opta.

Короткий зміст роботи: Дипломна робота присвячена розробці дротового розумного будинку на основі PLC Arduino Opta та програмування його в Arduino IDE та Arduino Cloud

КЛЮЧОВІ СЛОВА: SMART HOME, PLC, ARDUINO, ARDUINO OPTA, IOT, ARDUINO IDE, FINDER OPTA

ABSTRACT

Text part of the master's qualification work:
pages, pictures, table, sources.

The purpose of the work - Design and Implementation of a Wireless Smart Home System based on Arduino Opta PLC

Object of research – The smart home system based on the Arduino Opta programmable logic controller.

Subject of research – Aspects related to the development of a wireless smart home, considering the unique features and capabilities of the Arduino Opta platform.

Summary of the work - This thesis is dedicated to the development of a wireless smart home utilizing the PLC Arduino Opta and programming it using Arduino IDE and Arduino Cloud.

KEYWORDS: SMART HOME, PLC, ARDUINO, ARDUINO OPTA, IOT, ARDUINO IDE, FINDER OPTA

ВІДГУК РЕЦЕНЗЕНТА
на кваліфікаційну магістерську роботу

здобувача вищої освіти Білецький Ігор Сергійович
(*прізвище, ім'я, по батькові*)

на тему «Проектування та реалізація системи розумного будинку на основі Arduino»

Актуальність.

Робота є актуальною у зв'язку зі зростанням інтересу до Інтернету Речей та проектування дротового розумного будинку. PLC Arduino Opta поєднує у собі надійність промислового рівня та простоту використання Arduino.

Позитивні сторони.

1. Arduino Opta є новим рішенням, для промислової автоматизації, проте у цій дипломній роботі, Opta доволі зрозуміло описується як рішення домашньої автоматизації
2. Порівняння різних PLC на основі Arduino.
3. Розглянута інтеграція та керування PLC та підключеного до нього обладнання за допомогою Arduino Cloud.

Недоліки.

1. Не розглянуті недоліки Arduino Opta.
2. Не розглянутий метод програмування Arduino Opta за допомогою Arduino PLC.

Відзначені зауваження не впливають на загальну позитивну оцінку кваліфікаційної магістерської роботи.

Висновок: *кваліфікаційна магістерська робота заслуговує оцінку "відмінно," а здобувач _____ заслуговує присвоєння кваліфікації:*

Рецензент:
науковий ступінь, вчене звання

підпис *Ім'я, ПРІЗВИЩЕ*

ЗМІСТ

1 IoT ТА ЕКОСИСТЕМА ARDUINO	Помилка! Закладку не визначено.
1.1 Сімейство мікроконтролерів Arduino	Помилка! Закладку не визначено.
1.2 Платформа arduino у системі розумного будинку	Помилка! Закладку не визначено.
1.3 PLC на основі Arduino	Помилка! Закладку не визначено.
2 ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРІВ ARDUINO ЗА ДОПОМОГОЮ ARDUINO IDE	Помилка! Закладку не визначено.
2.1 Arduino PLC IDE	Помилка! Закладку не визначено.
2.2 Дротові протоколи комунікації	Помилка! Закладку не визначено.
3 СИСТЕМА ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ З PLC ARDUINO ОРТА	Помилка! Закладку не визначено.
3.1 Управління енергією за допомогою Opta	Помилка! Закладку не визначено.
3.2 Рекомендації щодо створення дротового розумного будинку на основі PLC	Помилка! Закладку не визначено.
ВИСНОВКИ	Помилка! Закладку не визначено.
ПЕРЕЛІК ПОСИЛАНЬ	76
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація).....	78

1 IoT ТА ЕКОСИСТЕМА ARDUINO

Інтернет речей (IoT) - це концепція мережі, що складається із взаємозв'язаних фізичних пристроїв, які мають вбудовані датчики, а також програмне забезпечення, що дозволяє здійснювати передачу і обмін даними між фізичним світом і комп'ютерними системами в автоматичному режимі, за допомогою використання стандартних протоколів.

У найпростішому розумінні, IoT - це мережа речей, які можуть спілкуватися між собою та з людьми за допомогою Інтернету. Ці речі можуть бути оснащені датчиками, які збирають інформацію про навколишнє середовище, або можуть виконувати певні дії, наприклад, включити світло або відчиняти двері.

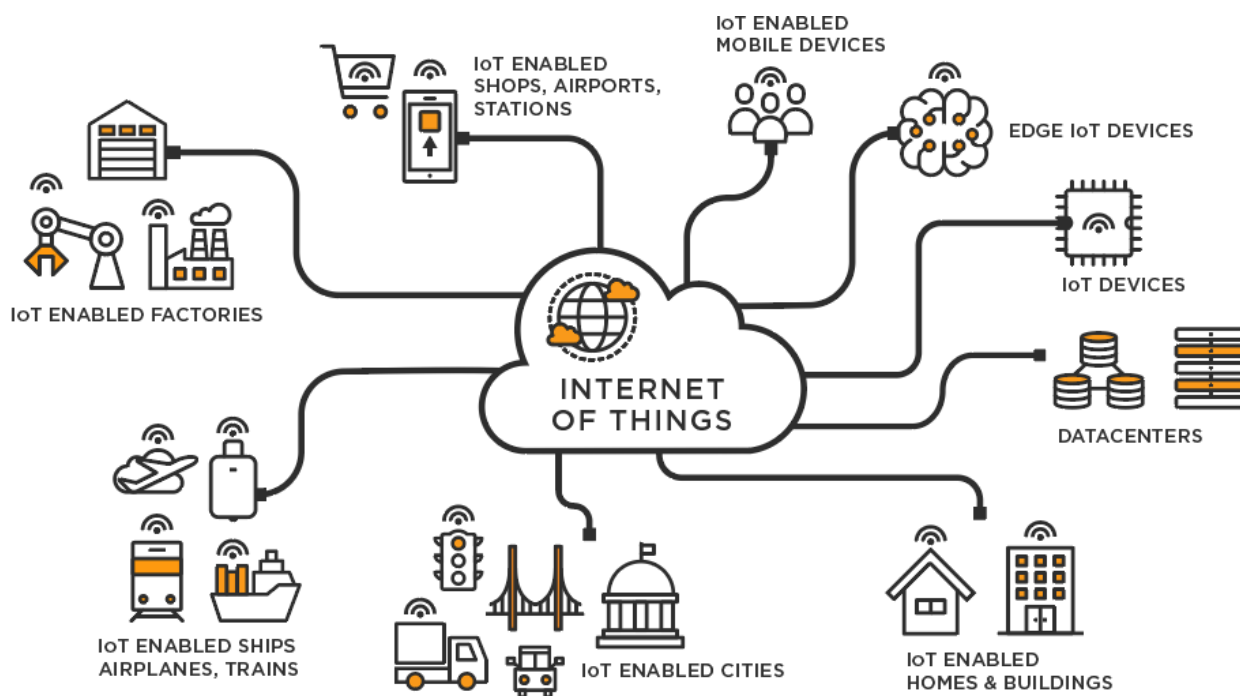


Рисунок 1.1 Графічне зображення Інтернету речей (IoT)

В основі IoT лежить Інтернет-протокол (IP) і протокол керування передачею (TCP). Ці стандарти та правила є основою для датчиків, пристроїв і систем для з'єднання з Інтернетом і один з одним. IoT обробляє дані з пристроїв і передає інформацію через дротові та бездротові мережі, включаючи Ethernet, Wi-Fi, Bluetooth, стільниковий зв'язок 5G і LTE, радіочастотну ідентифікацію (RFID) і зв'язок ближнього поля (NFC) [9]. Як правило, пристрої IoT підключаються до

шлюзів IoT або периферійних пристроїв, які збирають дані. Вони передають дані в хмарні обчислювальні середовища та з них, які зберігають і обробляють інформацію [10]. Широкий набір мережевих стандартів гарантує, що даними можна ділитися та досягати правильної «речі», таким чином з'єднуючи фізичний світ із цифровим.

У 1982 році кілька аспірантів факультету інформатики Карнегі-Меллона підключили торговельний автомат Coca-Cola до Інтернету заради розваги. Машина відображатиме на веб-сторінці свою температуру та різний запас газованої води в реальному часі. Цей проект є першим відомим прикладом IoT.

У 1997 р. британський технолог Кевін Ештон, співзасновник Auto-ID Center в Массачусетському технологічному інституті, почав досліджувати технологічну структуру, радіочастотну ідентифікацію (RFID), яка дозволить фізичним пристроям підключатися через мікročіпи та бездротові сигнали, і саме у своїй промові в 1999 році Ештон придумав фразу «Інтернет речей». Протягом кількох років смартфони, хмарні обчислення, прогрес у обчислювальній потужності та покращені алгоритми програмного забезпечення створили основу для збору, зберігання, обробки та обміну даними більш надійним способом. У той же час з'явилися складні датчики, які могли вимірювати рух, температуру, рівень вологості, напрямок вітру, звук, світло, зображення, вібрацію та багато інших умов, а також здатність точно визначити людину чи пристрій за допомогою геолокації. Ці розробки зробили можливою можливість спілкуватися як з цифровими пристроями, так і з фізичними об'єктами в реальному часі.

Останній звіт IoT Analytics « State of IoT—Spring 2023 » показує, що кількість глобальних підключень IoT зросла на 18% у 2022 році до 14,3 мільярда активних кінцевих точок IoT. У 2023 році IoT Analytics очікує, що глобальна кількість підключених пристроїв IoT зросте ще на 16% до 16,7 мільярда активних кінцевих точок. Хоча прогнозується, що зростання у 2023 році буде дещо нижчим, ніж у 2022 році, очікується, що кількість підключень пристроїв IoT продовжуватиме зростати протягом багатьох наступних років [16].

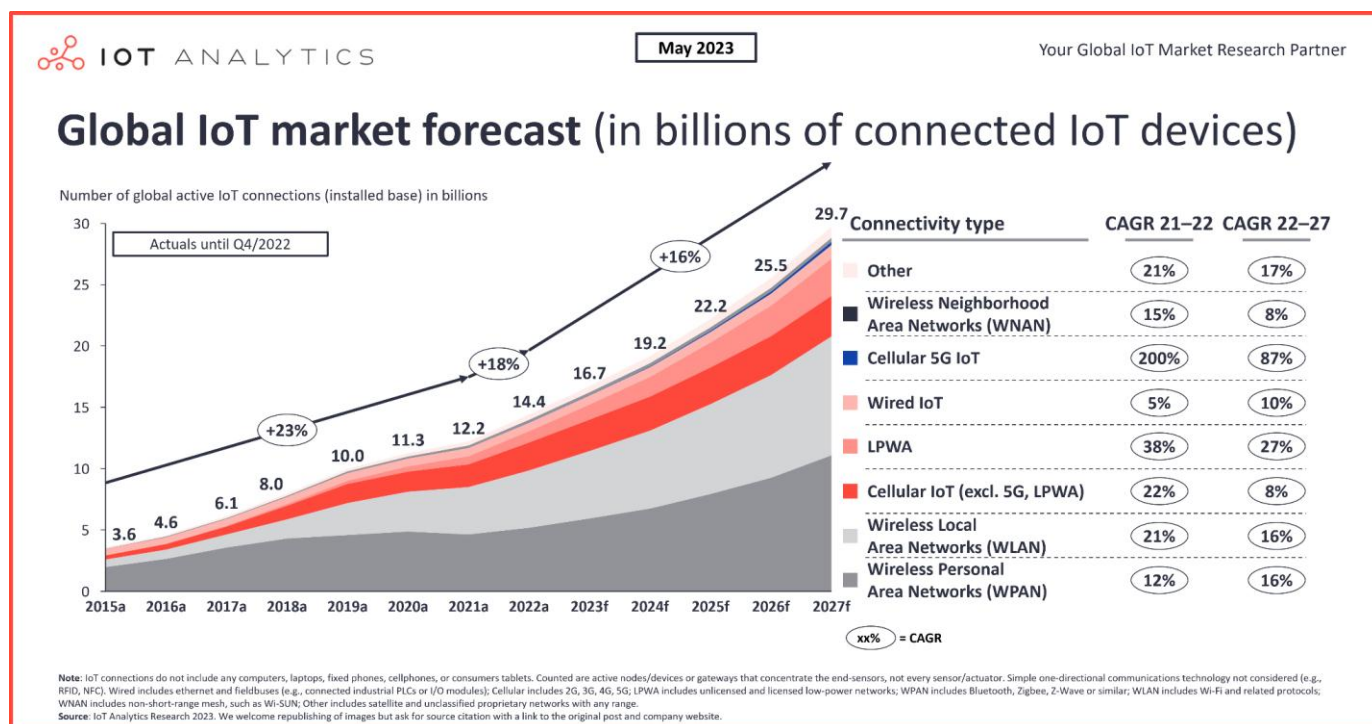


Рисунок 1.2 Прогноз кількості IoT пристроїв у світі

Хоча ідея IoT існує вже давно, сукупність останніх досягнень у низці різних технологій зробила її практичною.

- Доступ до недорогої сенсорної технології з низьким енергоспоживанням. Доступні та надійні датчики роблять технологію IoT можливою для більшої кількості виробників.
- Підключення. Велика кількість мережевих протоколів для Інтернету полегшує підключення датчиків до хмари та інших «речей» для ефективної передачі даних.
- Платформи хмарних обчислень. Збільшення доступності хмарних платформ дозволяє як компаніям, так і споживачам отримувати доступ до інфраструктури, необхідної для розширення, без необхідності фактично керувати нею.
- Машинне навчання та аналітика. Завдяки прогресу в машинному навчанні та аналітиці, а також доступу до різноманітних і величезних обсягів даних, що зберігаються в хмарі, компанії можуть збирати інформацію швидше та легше. Поява цих суміжних технологій продовжує розширювати межі IoT, а дані, створені IoT, також живлять ці технології.

- Розмовний штучний інтелект (ШІ). Прогрес у нейронних мережах привніс обробку природної мови (NLP) у пристрої IoT (такі як цифрові персональні помічники Alexa, Cortana та Siri) і зробив їх привабливими, доступними та життєздатними для домашнього використання [8].

IoT має величезний потенціал для розвитку. Він може змінити багато сфер нашого життя, зробивши її більш зручною, ефективною та безпечною. Ось деякі з перспектив розвитку IoT:

Розвиток розумних міст - використання IoT для створення більш безпечних, екологічно чистих та енергоефективних міст.

Розвиток розумного сільського господарства - використання IoT для підвищення ефективності сільського господарства та захисту навколишнього середовища.

Розвиток розумного транспорту - використання IoT для підвищення безпеки та ефективності транспорту.

Розвиток розумного здоров'я - надання персоналізованих медичних послуг.



Рисунок 1.3 Приклад напрямків використання IoT

Роль IoT в концепції розумного будинку

Розумний дім – це будинок з підключеними до Інтернету приладами, якими можна дистанційно керувати за допомогою планшета чи смартфона [17]. Він використовує розумні пристрої, такі як смарт-телевізори, розумні термостати, кондиціонери та навіть робот-пилосос. Потім вони з'єднуються в одну мережу за допомогою провідних або бездротових систем, таких як Zigbee, Wi-Fi, Bluetooth і NFC тощо.

Використовуючи технологію Інтернету речей (IoT), ваші інтелектуальні пристрої можуть спілкуватися та обмінюватися даними в реальному часі один з одним. Це дозволяє пристроям виконувати заплановані та автоматизовані завдання:

Контроль і моніторинг. Значний прогрес у контролі та моніторингу домашніх процесів став можливим завдяки Інтернету речей (IoT). Наприклад, ваш холодильник може автоматично додати молоко до вашого списку покупок або попередити вас, якщо ваш йогурт зіпсується через два дні.

Економія вартості та енергії. Програми IoT можуть надавати обширну статистику, яка допоможе вам покращити використання енергії та зменшити витрати шляхом збору та обробки даних про використання енергії та продуктивність смарт-пристроїв. Крім того, ви можете запрограмувати свою систему домашнього комплекту на автоматичне вимикання гаджетів, які не використовуються, для економії енергії, коли вдома нікого немає.

Вплив на навколишнє середовище. Енергозбереження може зменшити ваші витрати, а також принести користь навколишньому середовищу та сприяти більш екологічному способу життя.



Рисунок 1.4 приклад розумного будинку

Краща безпека. Завдяки повному контролю над усім у вашому домі та поза ним, система безпеки розумного будинку є чудовим інструментом для захисту ваших активів. Щоб повідомити вас про проблему, камери, дверні замки, датчики руху, диму та інші датчики співпрацюють. Ви можете стежити за тим, що відбувається всередині та за межами вашого дому, будь то вдома, на роботі чи під час відпочинку на пляжі на півдорозі світу.

Комфорт. Люди завжди прагнули до більшого комфорту та зручності, і впровадження технології Інтернету речей у розумних будинках є, мабуть, найважливішим кроком у цьому напрямку. Оскільки тепер багато побутових функцій можна виконувати незалежно, зберігаючи загальний контроль, він чудово спрощує та покращує ваше життя. Домашні гаджети IoT передають дані у мережі за допомогою датчиків. Деякі програми розумного дому в Інтернеті речей – це автоматичні вимикачі світла, охоронна сигналізація та звукові системи з голосовим керуванням.

1.1 Сімейство мікроконтролерів Arduino

Плата Arduino була розроблена в Ivrea Interaction Design Institute для студентів, які не мають досвіду в електроніці та програмуванні [1]. Ця плата

почала змінюватися, щоб адаптуватися до нових вимог і викликів, відокремлюючи свою присутність від простих 8-бітних плат до продуктів для додатків IoT (Інтернет речей), 3D-друку, пристроїв для носіння та вбудованого середовища. Усі дошки мають повністю відкритий вихідний код, що дозволяє користувачам створювати їх окремо та нарешті адаптувати під свої потреби. Протягом багатьох років різні типи плат Arduino використовувалися для створення тисяч проектів, від побутових об'єктів до складних наукових приладів. Міжнародна спільнота дизайнерів, художників, студентів, програмістів, любителів та експертів об'єдналася навколо цих пристроїв з відкритим кодом, їхні пожертви створили неймовірну кількість доступних знань, які можуть бути надзвичайно корисними як для новачків, так і для спеціалістів.

Які бувають типи плат Arduino?

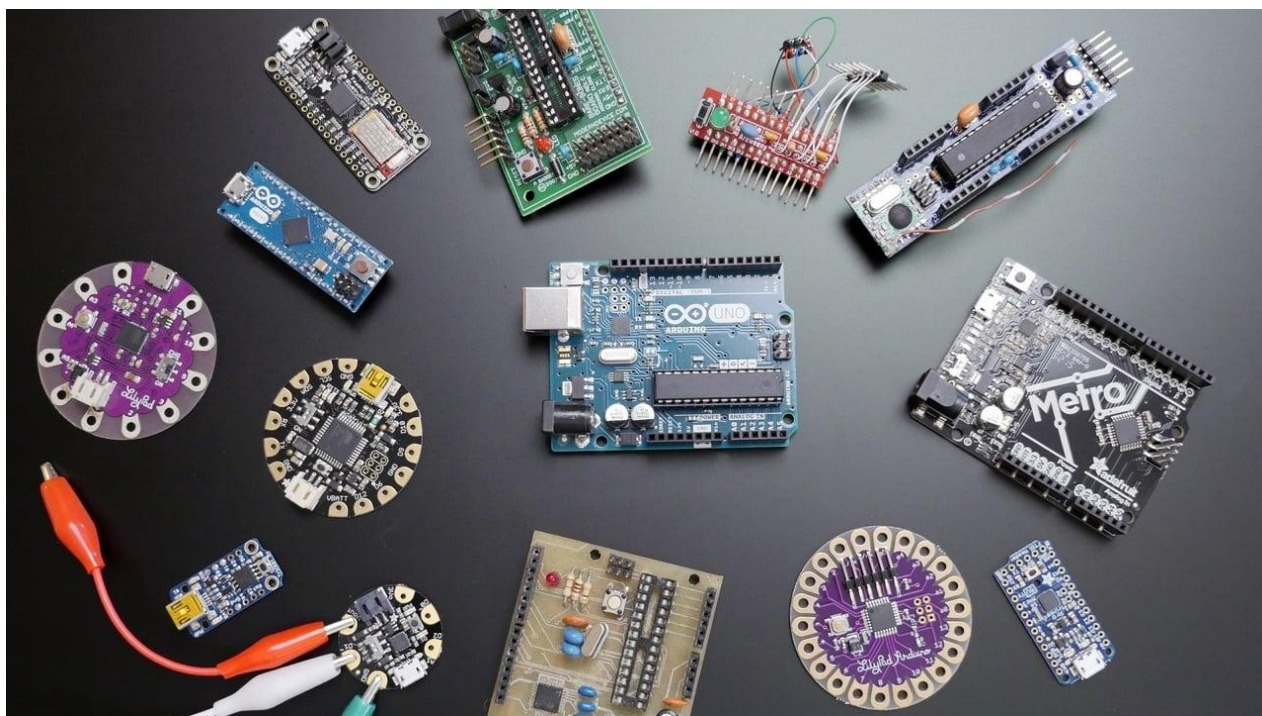


Рисунок 1.5 Типи плат Arduino

На відміну від більшості попередньо програмованих плат, Arduino не потребує окремої частини апаратного забезпечення, щоб запрограмувати новий код на платі, ви можете просто використовувати кабель USB. Крім того, Arduino IDE використовує базову версію C++, що спрощує вивчення програми.

Arduino Uno R3

Arduino Uno R3 — це популярна плата серед електроніки DIY, яка пропонує такі функції, як 14 цифрових контактів входу/виводу, 6 аналогових контактів і роз'єм ICSP (In-Circuit Serial Programming) [26].

Він працює на мікросхемі ATmega328P 16 МГц, що забезпечує живлення напруги до 5 В до підключених компонентів. Хоча він використовує роз'єм USB-B для комп'ютерного інтерфейсу, це не означає, що проекти, які потребують передових навичок, не можуть бути створені з ним.

Основні характеристики включають можливість пам'яті 2 КБ SRAM, 32 КБ пам'яті флеш-пам'яті та мікросхему EEPROM ємністю 1 КБ разом з UART і можливостями зв'язку I2C SPI, які також можна замінити у разі будь-якої проблеми. Це чудовий вибір для тих, хто хоче почати займатися електронікою та програмуванням своїми руками. Його можна використовувати як для простих, так і для складних проектів.

Arduino Uno є найбільш часто використовуваною платою, і це стандартна форма, крім усіх існуючих плат Arduino. Ця дошка дуже корисна для новачків.

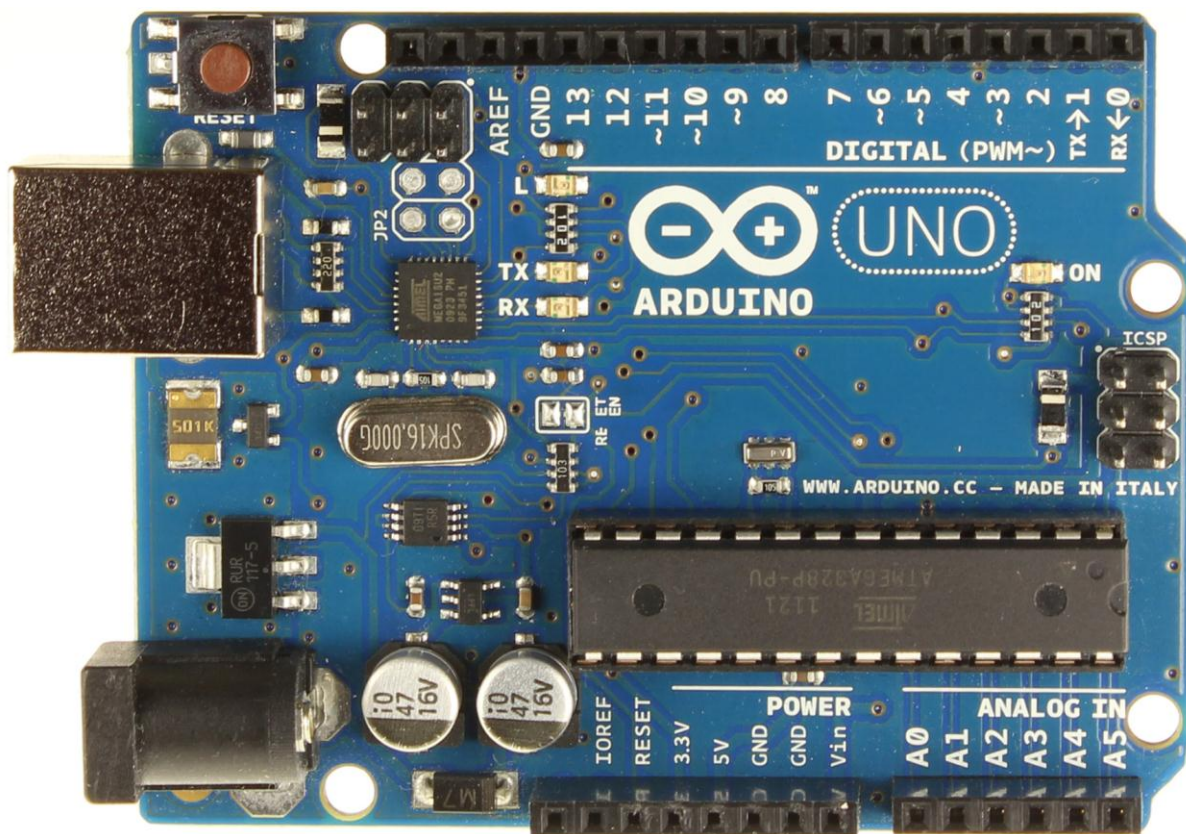


Рисунок 1.6 Arduino Uno (R3)

Arduino Nano

Arduino Nano дуже популярний серед розробників завдяки своєму невеликому розміру, але значним можливостям. Він має багато тих самих функцій, що й UNO, але в меншій формі, що робить його чудовим для використання на макетних платах та інших обмежених місцях. Основні відмінності від більшої моделі:

Контроллер немає вбудованого роз'єму живлення постійного струму. Замість цього він покладається на міні-порт USB, який також обслуговує чіпи моста UART із виділеним конвертером FT232RL FTDI. Це означає, що чіпсет ATmega16U2 не потрібен.

Ця комбінація забезпечує потужний контроль за скромною ціною та мінімальними вимогами до місця, дозволяючи творчому дизайну бути в межах досяжності.

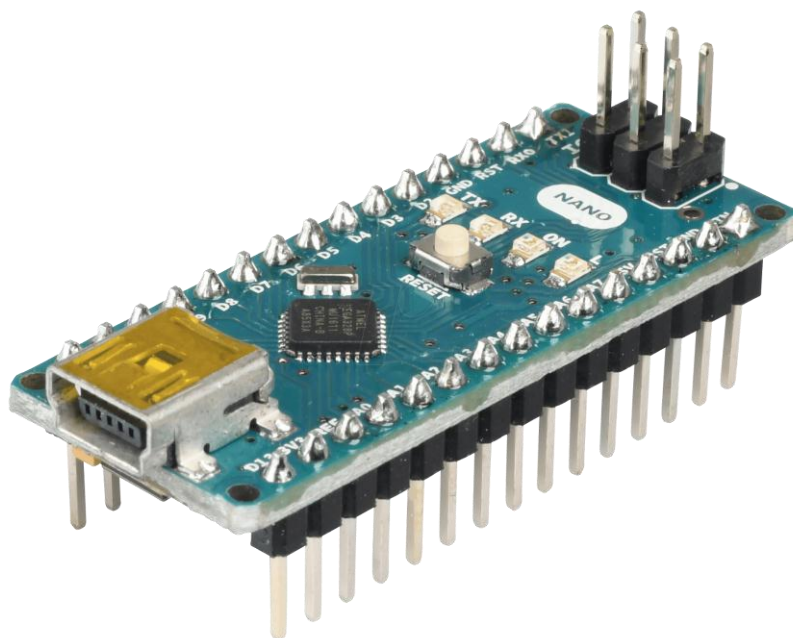


Рисунок 1.7 Arduino Nano

Arduino Micro

Arduino Micro пропонує функції, подібні до Leonardo, але значно менший і має лише 12 аналогових контактів введення/виведення разом із 20 цифровими.

Ця дошка має лише 18 мм у ширину та 48 мм у довжину. Це одна з найменших плат, коли-небудь створених Arduino. Ви можете створити клавіатуру,

мишу чи інші HID-пристрої, які вимагають мінімум місця, за допомогою цієї плати Arduino.

Він оснащений мікроконтролером ATmega32U4, що містить 2,5 КБ SRAM, 32 КБ флеш-пам'яті та 1 КБ EEPROM для зберігання даних, а також протоколи зв'язку UART, I2C і SPI. Він також має спеціальні вбудовані функції, такі як підключення HID.

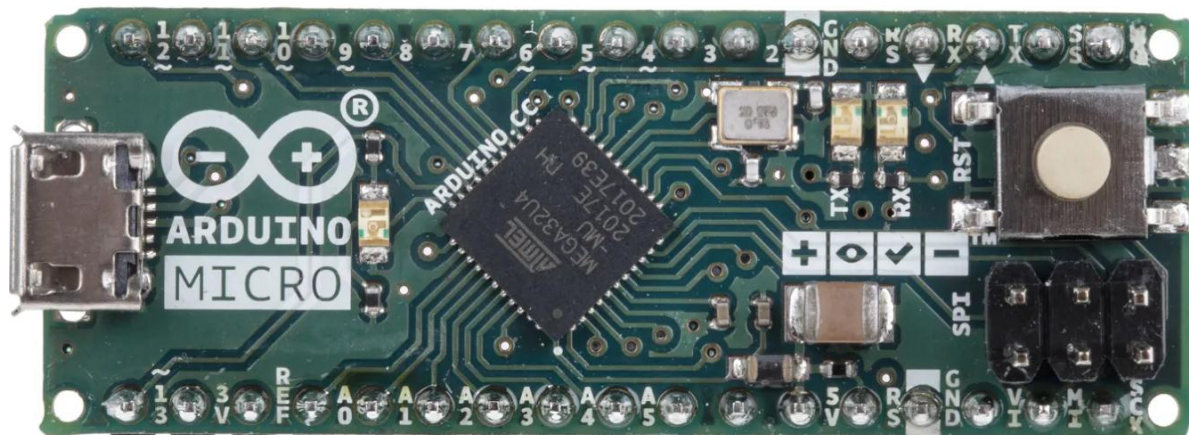


Рисунок 1.8 Arduino Micro

Arduino Due

Arduino Due є більш потужним оновленням популярної Arduino Mega з 54 цифровими входами/виходами та 12 аналоговими входами.

Ця плата, що працює лише на 3,3 В, та частоті 84 МГц і має процесор Atmel SAM3x8E Arm Cortex-M3, а також інтерфейси USB OTG, DAC і JTAG для додаткових можливостей підключення.

Важливо зазначити, що штифти вводу-виводу цього пристрою не витримують напруги 5 В, тому пам'ятайте про це, починаючи роботу над проектом.

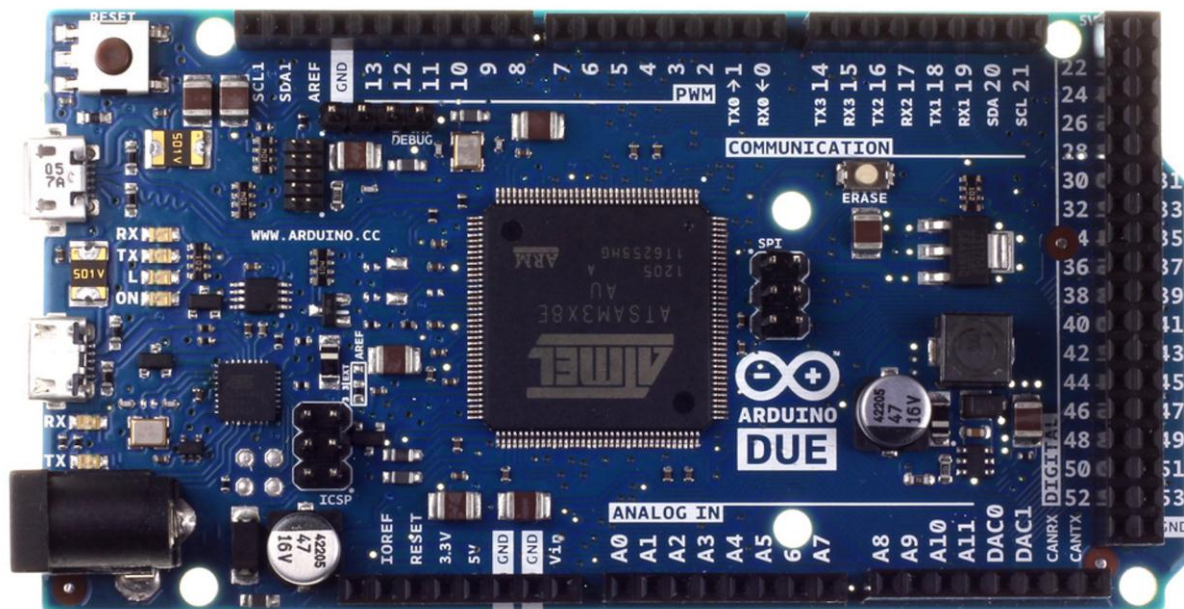


Рисунок 1.9 Arduino Due

Arduino Mega (R3)

Arduino Mega схожий на старшого брата UNO. Він містить багато цифрових контактів вводу/виводу (з них 14 контактів можна використовувати як PWM o/ps), 6 аналогових входів, кнопку скидання, роз'єм живлення, USB-з'єднання та кнопку скидання. Він містить усе необхідне для підтримки мікроконтролера; просто підключіть його до комп'ютера за допомогою USB-кабелю та подайте джерело живлення, щоб розпочати роботу, використовуючи адаптер змінного струму в постійний струм або акумулятор. Величезна кількість контактів робить цю плату Arduino дуже корисною для проектування проектів, які потребують купи цифрових i/ps або o/ps, таких як багато кнопок. Перейдіть за цим посиланням, щоб дізнатися більше про плату Arduino Mega (R3).

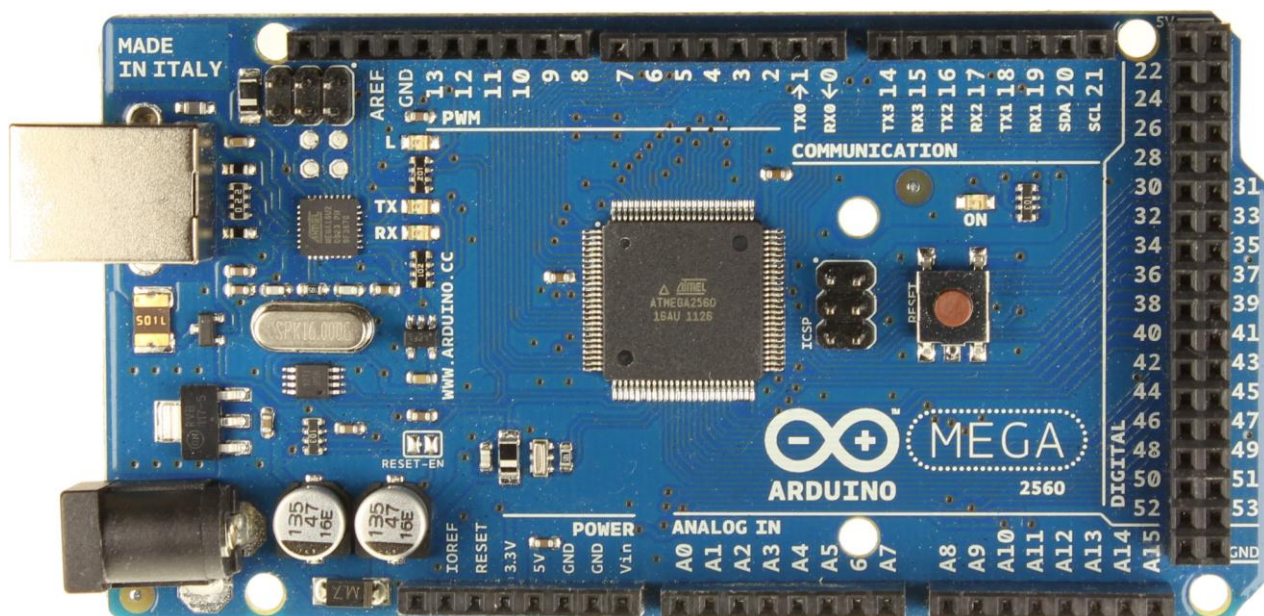


Рисунок 1.10 Плата Arduino Mega (R3)

Плата Arduino Leonardo

Першою платою розробки Arduino є плата Leonardo. Ця плата використовує один мікроконтролер разом із USB. Це означає, що це може бути дуже простим і дешевим. Оскільки ця плата безпосередньо обробляє USB, можна отримати програмні бібліотеки, які дозволяють платі Arduino слідувати клавіатурі комп'ютера, миші тощо.

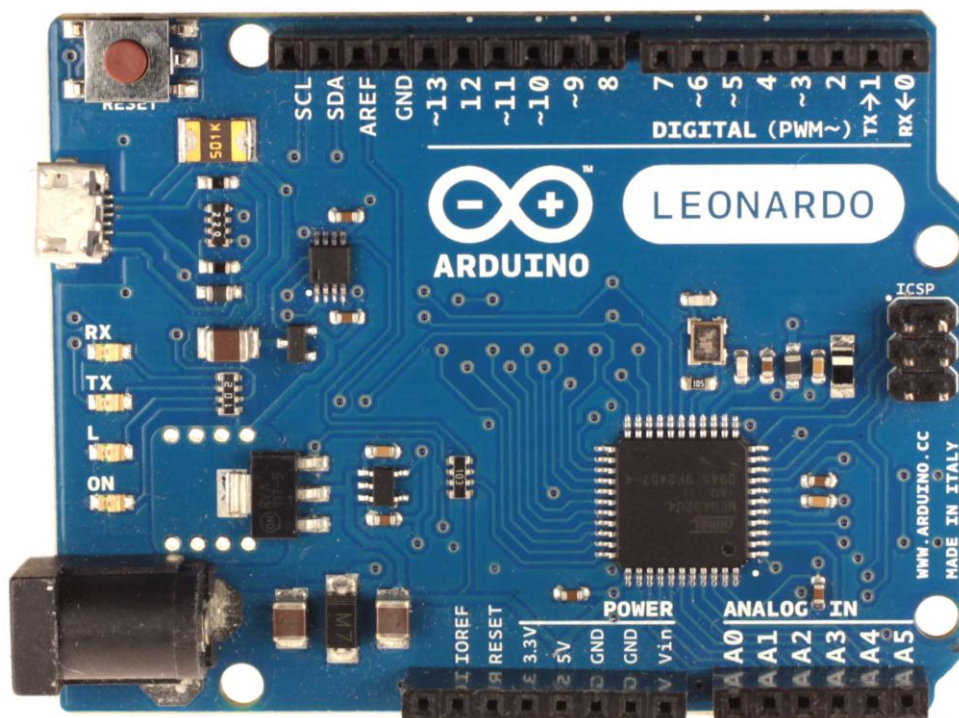


Рисунок 1.11 Плата Arduino Leonardo

Arduino Pro

Плата Arduino Pro Micro така ж, як і плата Arduino Mini, за винятком мікроконтролера ATmega32U4. Ця професійна мікрофонна плата містить контакти цифрового вводу/виводу-12, контакти широтно-імпульсної модуляції (ШИМ)-5, послідовні з'єднання Tx & Rx і 10-бітний АЦП.

Arduino Protenta

Arduino Protenta — це сімейство мікроконтролерів, розроблених для промислової автоматизації [2]. Portenta H7 одночасно виконує код високого рівня разом із завданнями в реальному часі. Конструкція включає два процесори, які можуть виконувати завдання паралельно [25]. Наприклад, можна виконувати скомпільований код Arduino разом із кодом MicroPython і мати обидва ядра для зв'язку одне з одним. Protenta пропонує широкий спектр функцій, включаючи:

- Двоядерний процесор STM32H747XI ARM Cortex-M7 + M4
- 16 МБ флеш-пам'яті
- 2 МБ оперативної пам'яті
- 100 контактів GPIO
- USB-C, Ethernet, Wi-Fi, Bluetooth
- Підтримка стандартних мов програмування ПЛК
- Інтеграція польової шини через Modbus TCP/RTU
- Безшовне підключення до хмари

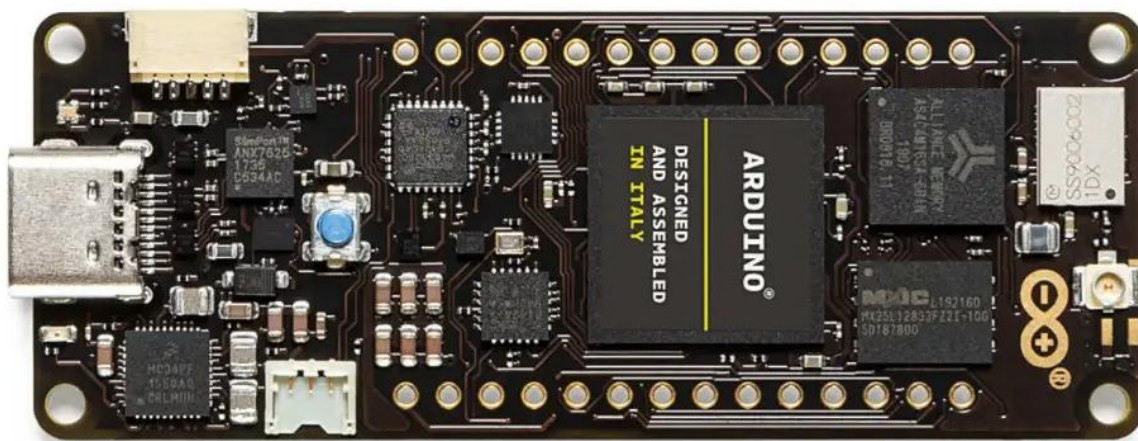


Рисунок 1.12 Arduino Protenta H7

Функціональність Portenta подвійна: вона може працювати як будь-яка інша плата вбудованого мікроконтролера або як головний процесор вбудованого комп'ютера. [3]

Основним процесором Н7 є двоядерний STM32H747, включаючи Cortex M7, що працює на частоті 480 МГц, і Cortex M4, що працює на частоті 240 МГц. Два ядра взаємодіють за допомогою механізму Remote Procedure Call, який дозволяє безперешкодно викликати функції іншого процесора. Обидва процесори спільно використовують усі вбудовані периферійні пристрої та можуть працювати.

Ймовірно, однією з найцікавіших функцій Portenta Н7 є можливість підключення зовнішнього монітора для створення власного спеціального вбудованого комп'ютера з інтерфейсом користувача. Це можливо завдяки вбудованому в процесор графічному процесору STM32H747, Chrom-ART Accelerator. Окрім графічного процесора, чіп містить спеціальний кодер і декодер JPEG.

Вбудований бездротовий модуль дозволяє одночасно керувати підключенням WiFi і Bluetooth. Інтерфейс WiFi може працювати як точка доступу, станція або як дворезимний одночасний AP/STA і може працювати зі швидкістю передачі до 65 Мбіт/с. Інтерфейс Bluetooth підтримує Bluetooth Classic і BLE. [12] Також можна підключити серію різних дротових інтерфейсів, таких як UART, SPI, Ethernet або I2C, як через деякі з роз'ємів у стилі MKR, так і через нову пару промислових 80-контактних роз'ємів Arduino.

1.2 Платформа arduino у системі розумного будинку

Arduino - це відкрите апаратне та програмне забезпечення для розробки електронних пристроїв. Головна ідея Arduino полягає в тому, щоб забезпечити

простий та доступний інтерфейс для створення проектів з використанням мікроконтролерів. Arduino дозволяє навіть людям без глибоких знань в електроніці та програмуванні створювати різноманітні електронні пристрої.

Основною складовою Arduino є плати, які містять мікроконтролер, вводи/виводи для підключення до інших пристроїв, а також інші елементи. Arduino підтримує широкий спектр сенсорів, актуаторів та інших електронних компонентів.

Програмування для Arduino виконується мовою C/C++, а Arduino IDE (Integrated Development Environment) - це середовище для написання, компіляції та завантаження програм на плату Arduino [18]. Велика спільнота користувачів та розробників сприяє обміну ідеями, кодом та проектами, що робить Arduino популярним серед ентузіастів DIY (зроби сам) та професійних розробників.

Arduino використовується у домашній автоматизації для створення різноманітних пристроїв та систем, які допомагають автоматизувати та контролювати певні аспекти домашнього середовища. Ось кілька прикладів використання Arduino в домашній автоматизації:

Освітлення: Arduino можна використовувати для створення систем автоматизованого освітлення. Наприклад, ви можете збудувати систему, яка вмикатиме світильники вдень, коли освітлення недостатнє, або регулювати яскравість освітлення в залежності від часу доби.

Термостат: Arduino можна використовувати для створення "розумного" термостата, який автоматично регулює температуру в приміщенні залежно від заданого розкладу чи змін зовнішнього середовища.

Системи безпеки: Arduino можна використовувати для створення системи безпеки, яка включає в себе датчики руху, системи відеоспостереження, або навіть розпізнавання відбитків пальців для доступу.

Автоматизовані штори та жалюзі: Arduino може керувати моторизованими системами управління шторами чи жалюзі, роблячи їх автоматичними, залежно від освітлення, температури або часу доби.

Системи поливу: Arduino можна використовувати для автоматизації систем поливу рослин, активуючи їх в залежності від вологості ґрунту або розкладу поливу.

Мультимедійний центр: Arduino можна використовувати для створення системи домашнього розваг, яка управляє аудіо- та відеообладнанням, а також освітленням та іншими аспектами комфорту.

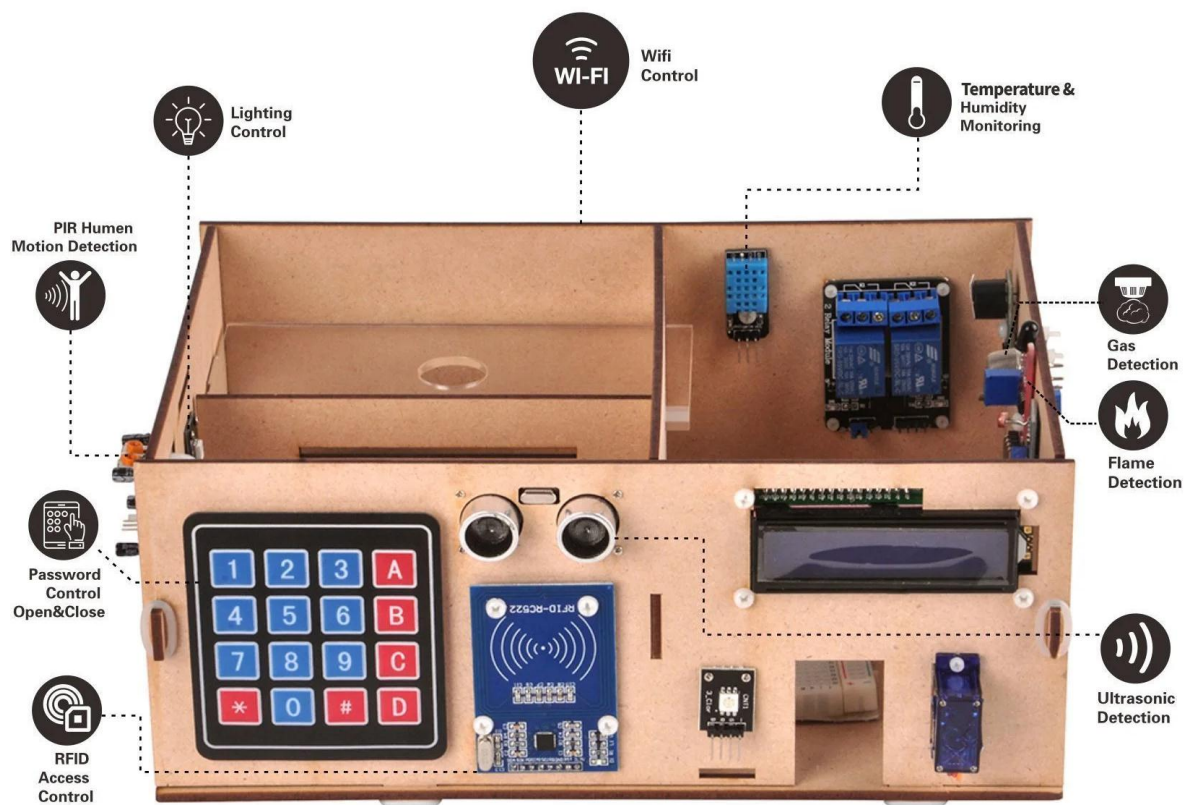


Рисунок 1.13 Приклад розумного будинку на основі Arduino [20]

Усі ці проекти базуються на розвитку програм, які програмуються на мові C/C++ в середовищі Arduino IDE, та використанні різноманітних сенсорів, модулів комунікації та актуаторів, які легко підключаються до Arduino-плати.

Використання PLC для потреб домашньої автоматизації

Домашня автоматизація - це процес використання технологій для автоматизації завдань у будинку. Вона може включати в себе такі речі, як управління освітленням, опаленням, безпекою та іншими системами.

Програмовані логічні контролери (PLC) є одними з найбільш широко використовуваних технологій для домашньої автоматизації [23]. Вони є

потужними і надійними пристроями, які можуть бути використані для контролю складних систем.

Приклади використання PLC для домашньої автоматизації

PLC можна використовувати для автоматизації широкого спектру завдань у будинку. Ось деякі приклади:

- **Управління освітленням:** PLC можна використовувати для автоматичного включення і вимкнення освітлення в будинку. Вони можуть також використовуватися для створення сценаріїв освітлення, таких як включення всіх вогнів у будинку при вході або вимкнення всіх вогнів при виході.

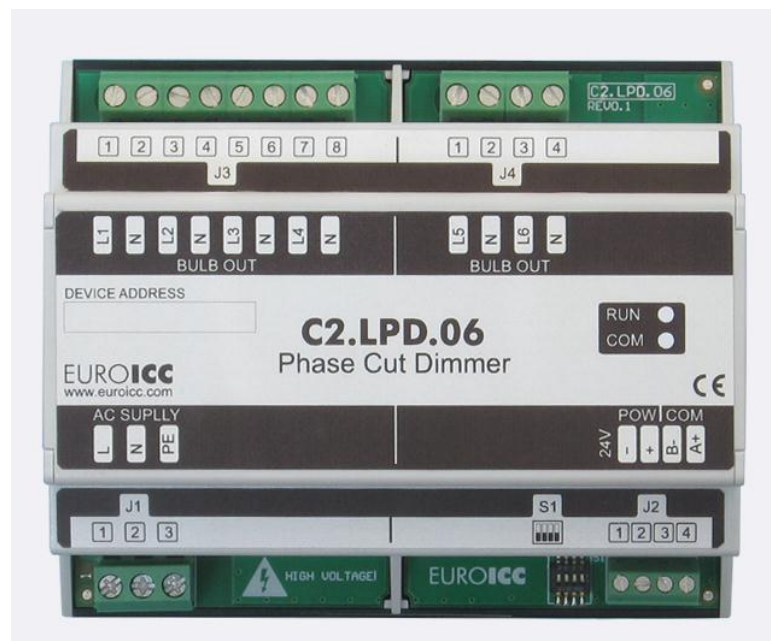


Рисунок 1.14 PLC для управління освітленням

- **Управління опаленням:** PLC можна використовувати для автоматичного регулювання температури в будинку. Вони можуть також використовуватися для створення сценаріїв опалення, таких як підвищення температури в будинку перед поверненням господарів додому.



Рисунок 1.15 PLC для управління опаленням

- Управління безпекою: PLC можна використовувати для автоматизації систем безпеки будинку, таких як відеодомофони, сигналізації та замки. Вони також можуть використовуватися для створення сценаріїв безпеки, таких як включення сигналізації при вході в будинок або відмикання замків при виході.



Рисунок 1.16 PLC для управління безпекою

- Управління іншими системами: PLC можна використовувати для автоматизації інших систем у будинку, таких як система поливу, система водопостачання та система вентиляції.

Ось деякі фактори, які будуть сприяти розвитку використання PLC для домашньої автоматизації:

- Зростання популярності розумних будинків: Розумні будинки - це будинки, в яких пристрої та системи взаємодіють між собою. PLC є ідеальними пристроями для керування такими системами.

- Розвиток технологій: Розвиток технологій, таких як штучний інтелект і машинне навчання, робить PLC ще більш потужними і функціональними.

- Зниження вартості PLC: PLC стають все більш доступними, що робить їх більш привабливими для широкого кола споживачів.

- Зростання популярності розумних будинків

Ось кілька конкретних прикладів сценаріїв використання PLC для домашньої автоматизації:

- Сценарій "Я вдома": Коли господарі входять у будинок, PLC може автоматично включити світло, підняти жалюзі та включити систему опалення.

- Сценарій "Я виходжу з дому": Коли господарі виходять з будинку, PLC може автоматично вимкнути світло, опустити жалюзі та вимкнути систему опалення.

- Сценарій "Сон": Коли господарі йдуть спати, PLC може автоматично вимкнути світло в спальні, включити нічник і запустити режим енергозбереження.

- Сценарій "Відпустка": Коли господарі йдуть у відпустку, PLC може автоматично вимкнути світло, опалення та інші системи.

PLC також можна використовувати для створення більш складних сценаріїв, які адаптуються до індивідуальних потреб користувачів. Наприклад, штучний інтелект може використовуватися для створення сценаріїв, які враховують такі фактори, як погодні умови, час доби та поточні потреби користувачів.

1.3 PLC на основі Arduino

PLC на основі Arduino - це пристрій, який поєднує в собі можливості PLC з відкритою платформою Arduino. Це дозволяє створювати недорогі і гнучкі рішення для домашньої автоматизації.

Portenta Machine Control

Це стандартний промисловий контролер soft-PLC, який може підключатися до широкого спектру зовнішніх датчиків і виконавчих механізмів за допомогою ізольованого цифрового входу/виходу, аналогового входу/виходу сумісного з 4-20 мА, 3 конфігурованих температурних каналів та виділеного роз'єму I2C.

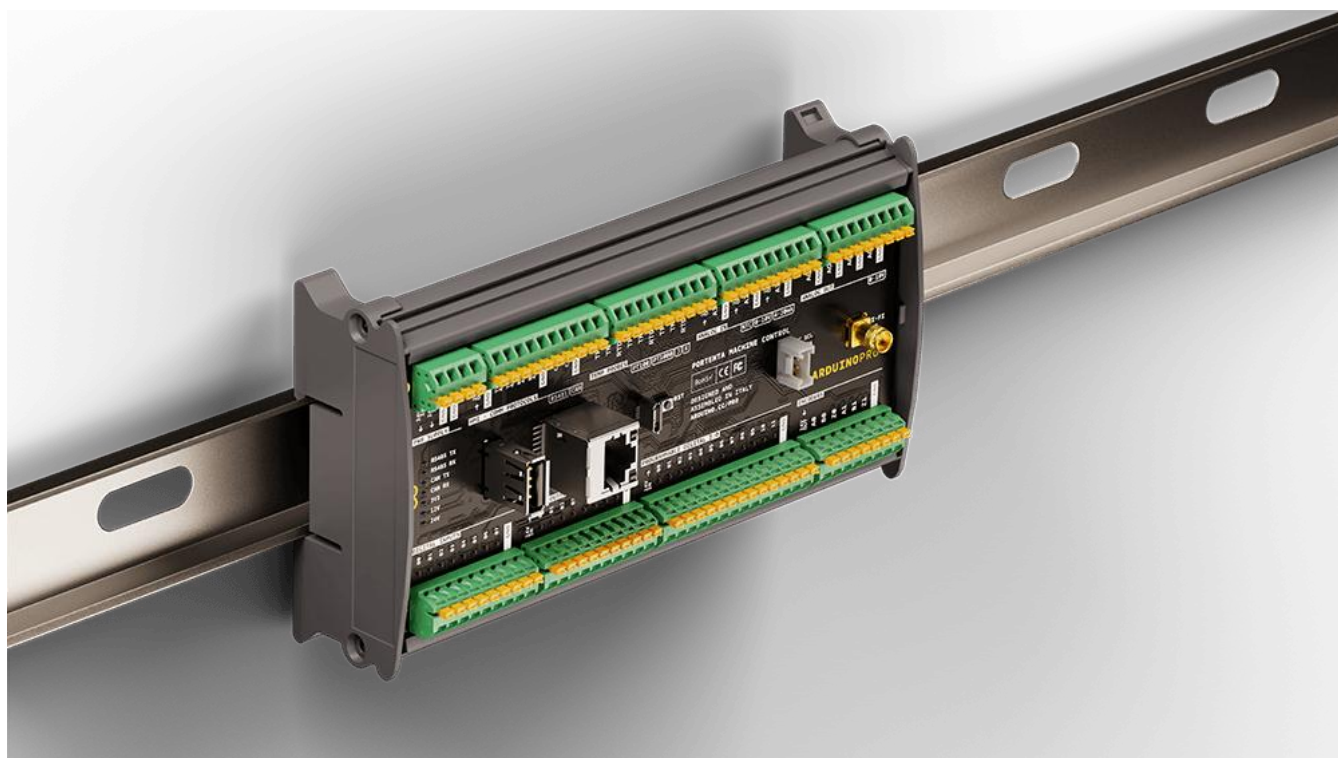


Рисунок 1.17 Portenta Machine Control

Ось деякі з ключових характеристик Portenta Machine Control:

Потужний процесор Machine Control працює на мікроконтролері Arduino Portenta H7, високонадійному дизайні, що працює в промислових температурних діапазонах (від -40°C до $+85^{\circ}\text{C}$) з двоядерною архітектурою, яка не вимагає зовнішнього охолодження.

Гнучкі можливості підключення: Portenta Machine Control має широкий спектр можливостей підключення, включаючи:

- Цифровий вхід/вихід: 16 цифрових вхідних/вихідних контактів, 8 з яких здатні до високого струму споживання або генерування
- Аналоговий вхід/вихід: 2 аналогових вхідних канали, сумісні з 4-20 мА
- Температурні канали: 3 конфігурованих температурних канали, з підтримкою різноманітних датчиків температури
- 1 виділений роз'єм I2C для підключення пристроїв I2C

Portenta Machine Control можна програмувати за допомогою середовища розробки Arduino, що робить його простим у використанні для всіх, хто має досвід програмування Arduino. Ось деякі з потенційних застосувань Portenta Machine Control:

Промислова автоматизація: контролер можна використовувати для керування широким спектром промислових автоматизованих застосувань, таких як: керування транспортом, упаковувальна техніка, робототехніка, системи HVAC,

Автоматизація будівель: Machine Control також можна використовувати для керування застосунками автоматизації будівель, такими як керування: освітленням, кліматом, доступом та системою безпеки.

Portenta Machine Control - це потужний і доступний варіант для проєктів домашньої автоматизації та DIY. Його можна використовувати для керування широким спектром пристроїв домашньої автоматизації, таких як: освітлення, терморегуляція, камери безпеки, розумні замки, тощо.

Загалом, Portenta Machine Control - це універсальний і потужний контролер soft-PLC, який добре підходить для широкого спектру застосувань.

Edge Control

Рішення для віддаленого моніторингу та керування, оптимізоване для зовнішнього середовища. Пульти керування Edge можна розташувати будь-де, він підходить для точного землеробства, інтелектуального сільського господарства та

інших застосувань, які потребують інтелектуального керування у віддалених місцях. Живлення може подаватись через сонячну панель або вхід постійного струму. Можна дистанційно керувати своїми додатками через хмару Arduino (або сторонні служби), використовуючи вибір варіантів підключення, за допомогою модемів 2G/3G/CatM1/NB-IoT, LoRa, Sigfox і WiFi, додавши будь-яку плату MKR.



Рисунок 1.18 Portenta Machine Control

Arduino Edge Control здатний підключати датчики та приводи. Крім того, він має можливість забезпечити моніторинг у реальному часі всього процесу, тим самим зменшуючи ризики, пов'язані з виробництвом. Датчики можуть збирати дані в режимі реального часу. Після надсилання до хмари Arduino ланцюжок створення вартості даних стає цінною аналітикою, яка підтримує бізнес-процеси на різних рівнях.

Conrollino

CONTROLLINO — це програмований логічний контролер (PLC) на основі мікроконтролерів (MCU) ATmega328 і Atmega2560, який поєднує в собі гнучкість і відкритий вихідний код екосистеми Arduino з безпекою та надійністю промислових ПЛК.

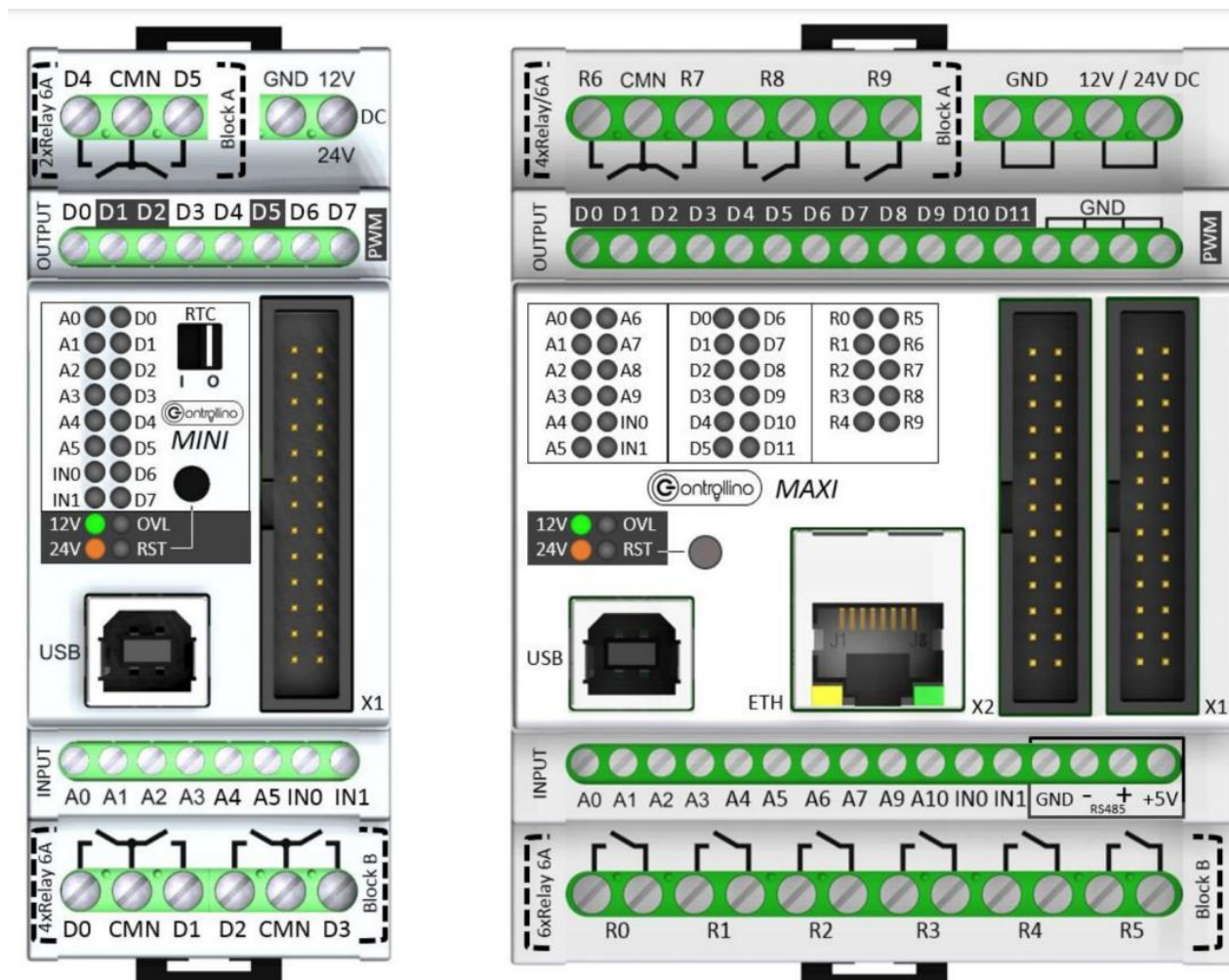


Рисунок 1.19 PLC Arduino Conrollino

Основні характеристики PLC Arduino Conrollino:

Програмування: CONTROLLINO програмується за допомогою Arduino IDE, що робить його доступним для широкого кола розробників.

Входи/виходи: CONTROLLINO має широкий набір входів/виходів, включаючи цифрові, аналогові, дискретні та аналогові вводи/виводи.

Периферія: CONTROLLINO має широкий спектр периферійних пристроїв, включаючи таймери, подільники частоти, аналогові-цифрові перетворювачі та цифрові-аналогові перетворювачі.

Контролери Conrollino доступні в різних конфігураціях, що відрізняються набором входів/виходів і периферії. Наприклад, Conrollino Mini має 10 цифрових і 2 аналогових входів/виходів, а Conrollino Mega має 54 цифрових і 12 аналогових входів/виходів.

M-Duino

M-duino - це промисловий Arduino для промислової автоматизації, керування процесами та збору даних. Цей програмований логічний контролер має 38 входів/виходів. Він також містить кілька комунікаційних портів, які забезпечують більшу гнучкість і контроль. Сімейство M-DUINO пропонує можливість розширення до 127 модулів через I2C, що означає, що ви можете мати до 7100 входів/виходів у з'єднаннях Master-Slave, додатково до датчиків.

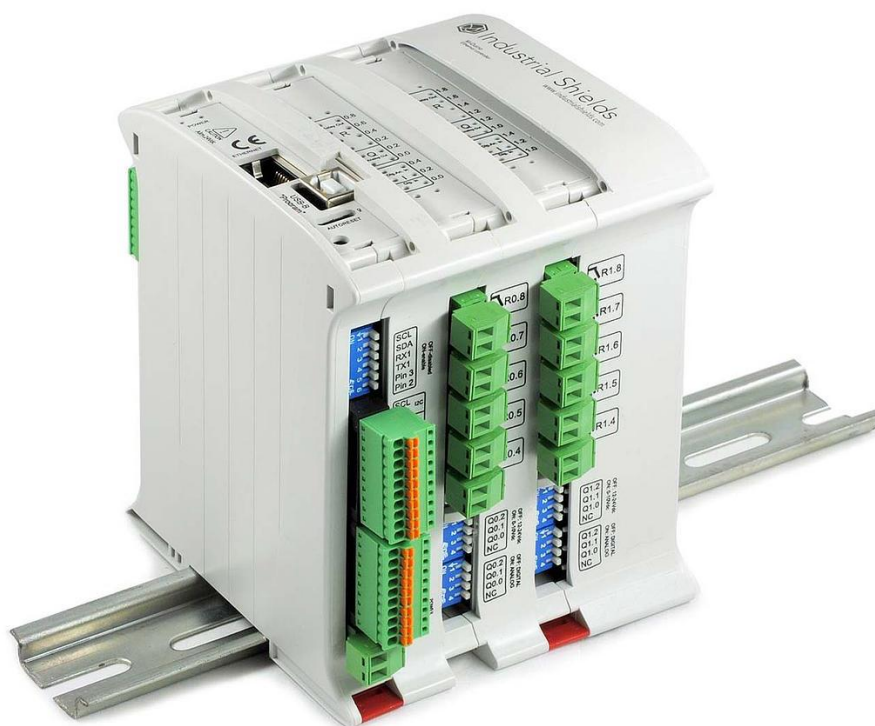


Рисунок 1.20 Arduino Opta

ОПТА — це назва абсолютно нової серії промислових розумних реле або програмованих логічних реле, які ідеально підходять для легкого створення інтелектуальної логіки для промислових, OEM-проектів або проектів автоматизації будівель. Ці пристрої можна вільно програмувати як традиційними мовами IEC61131-3 (Ladder, FBD тощо), так і мовою Arduino IDE з відкритим кодом [4].

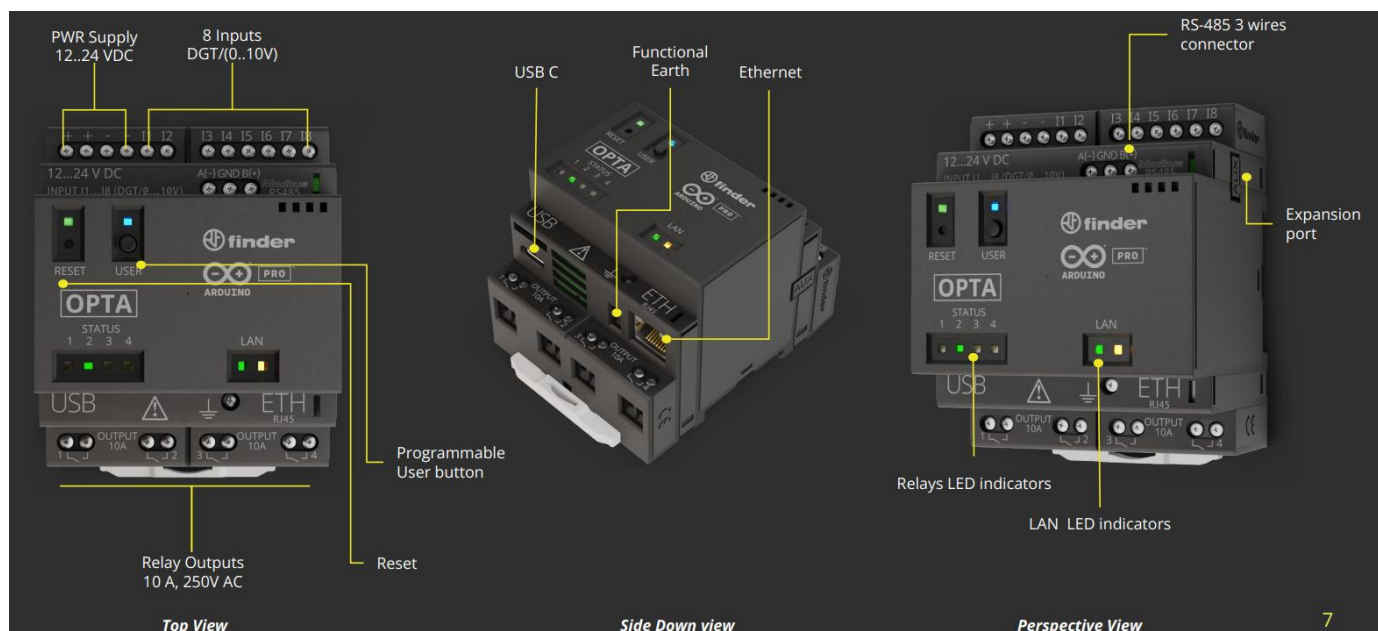


Рисунок 1.21 Arduino Opta

Його потужний двоядерний мікроконтролер STM32H747XI Cortex-M7 +M4 MCU дозволяє користувачам виконувати контроль, моніторинг і реалізацію додатків для прогнозованого обслуговування в реальному часі [5].

Захищений і надійний за своєю конструкцією, він підтримує OTA оновлення прошивки та забезпечує безпеку даних від апаратного забезпечення до хмари завдяки вбудованому елементу безпеки та сумісності зі стандартом X.509.

І все це зберігає фірмову легкість розгортання Arduino Pro у виробництві завдяки широкому спектру доступних бібліотек програмного забезпечення та ескізів Arduino.

І останнє, але не менш важливе: різноманітні варіанти підключення дозволяють без зусиль тримати все під контролем за допомогою інформаційних панелей у реальному часі в поєднанні з інтуїтивно зрозумілою хмарою Arduino (або сторонніми службами).

Opta доступна в трьох варіантах, тому ви можете вибрати найкращий варіант для свого проекту. Opta WiFi має вбудований Ethernet і порт програмування USB-C, напівдуплексний інтерфейс RS485 і підключення Wi-Fi/Bluetooth Low Energy.

2 ПРОГРАМУВАННЯ МІКРОКОНТРОЛЕРІВ ARDUINO ЗА ДОПОМОГОЮ ARDUINO IDE

Інтегроване середовище розробки Arduino - або програмне забезпечення Arduino (IDE) - містить текстовий редактор для написання коду, область повідомлень, текстову консоль, панель інструментів із кнопками для загальних функцій і ряд меню. Він підключається до апаратного забезпечення Arduino для завантаження програм і обміну даними з ними [27].

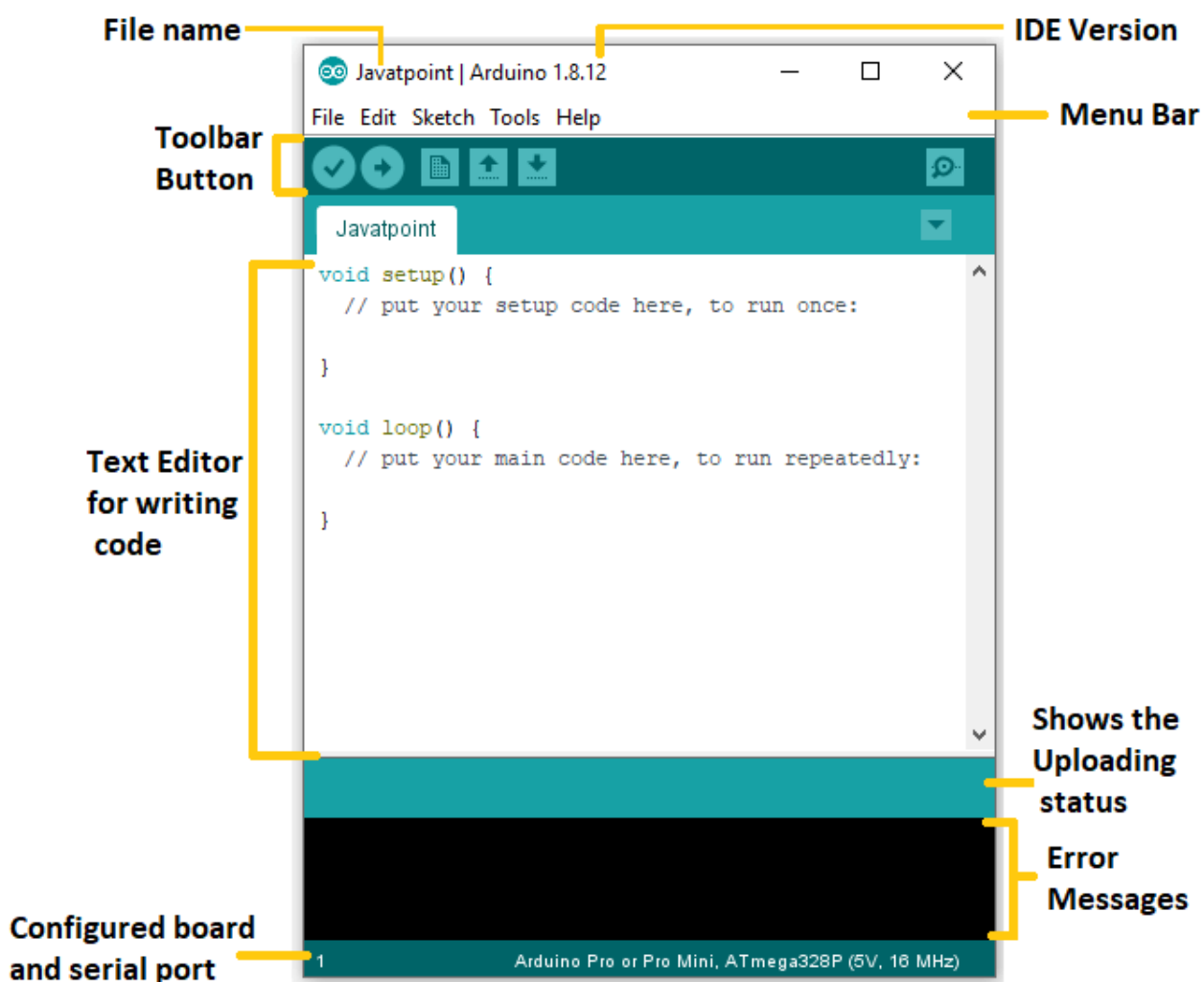


Рисунок 2.1 Arduino IDE

На панелі інструментів відображаються такі піктограми як: Створити, Відкрити, Зберегти, Завантажити та Перевірити. Це показано нижче:

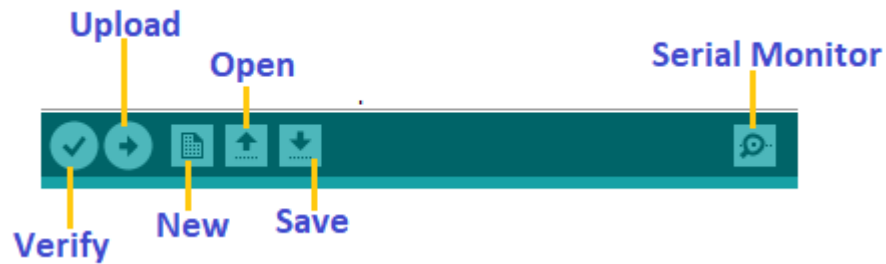


Рисунок 2.2 Панель інструментів Arduino IDE

Коли ми натискаємо кнопку «Файл» на панелі меню, з'являється розкритий список. Це показано нижче:

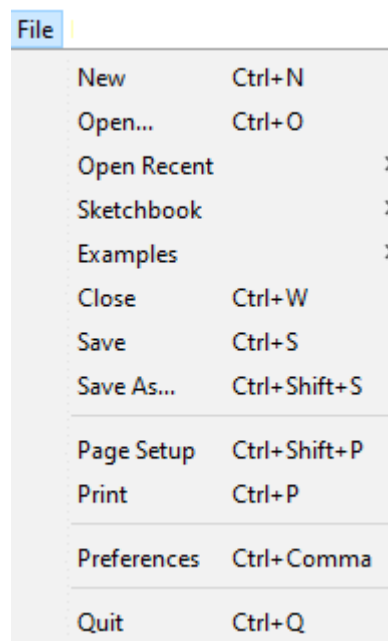


Рисунок 2.3 Меню файл Arduino IDE

Serial Monitor є важливим інструментом для створення проектів за допомогою Arduino. Його можна використовувати як інструмент налагодження, тестування концепцій або для прямого зв'язку з платою Arduino.

Піктограма лупи у верхній частині екрану відкриває доступ до послідовного монітору. Він зчитує данні мікроконтролера, через COM порт, або через безпроводний протокол.

Кнопка «Завантажити» компілює та запускає наш код, написаний на екрані. Далі він завантажує код на підключену плату. Перш ніж завантажити ескіз, ми повинні переконатися, що вибрано правильну плату та порти.

Коли ми натискаємо кнопку Інструменти на панелі меню, з'являється розкритий список. Це показано нижче:

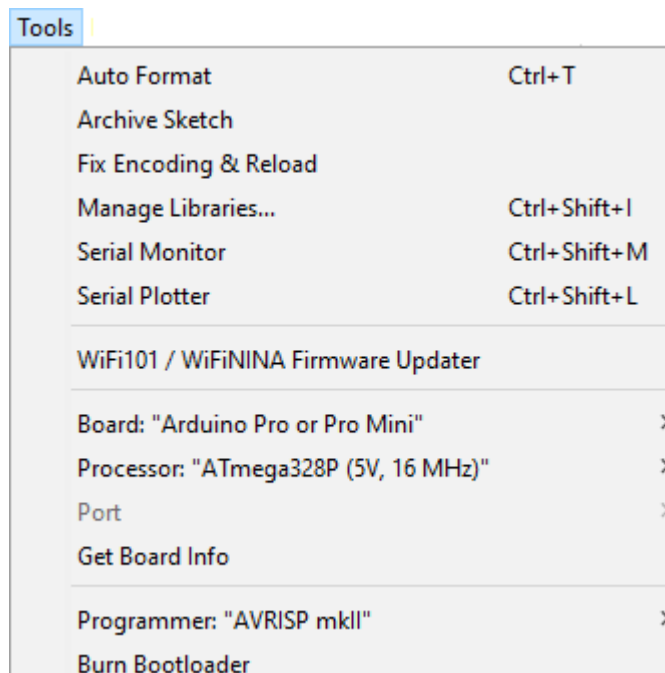


Рисунок 2.4 Меню інструменти Arduino IDE

Кнопка Auto Format використовується для форматування написаного коду. Наприклад, розбивання відкритих і закритих фігурних дужок у коді.

Копія поточного ескізу або коду архівується у форматі .zip. Каталог архіву такий самий, як і ескіз.

Кнопка виправити кодування та перезавантажити використовується, щоб виправити невідповідність між картами символів операційної системи та кодуванням карт символів редактора.

Програми, написані за допомогою програмного забезпечення Arduino (IDE), називаються ескізами. Ці ескізи написані в текстовому редакторі та збережені з розширенням файлу .ino. Редактор має функції вирізання/вставлення та пошуку/заміни тексту.

Область повідомлень надає відгук під час збереження та експорту, а також відображає помилки. На консолі відображається текстовий вихід програмного забезпечення Arduino (IDE), включаючи повні повідомлення про помилки та іншу інформацію. У нижньому правому куті вікна відображається налаштована плата та послідовний порт. Кнопки на панелі інструментів дозволяють перевіряти та

завантажувати програми, створювати, відкривати та зберігати ескізи, а також відкривати серійний монітор.

Бібліотеки надають додаткову функціональність для використання в ескізах, наприклад, робота з обладнанням або маніпулювання даними. Щоб використати бібліотеку в ескізі, виберіть її в меню «Ескіз» > «Імпортувати бібліотеку». Це вставить один або кілька операторів `#include` у верхній частині ескізу та компілює бібліотеку з вашим ескізом. Оскільки бібліотеки завантажуються на дошку разом із вашим ескізом, вони збільшують обсяг місця, який він займає. Якщо ескіз більше не потребує бібліотеки, просто видаліть його оператори `#include` у верхній частині коду.

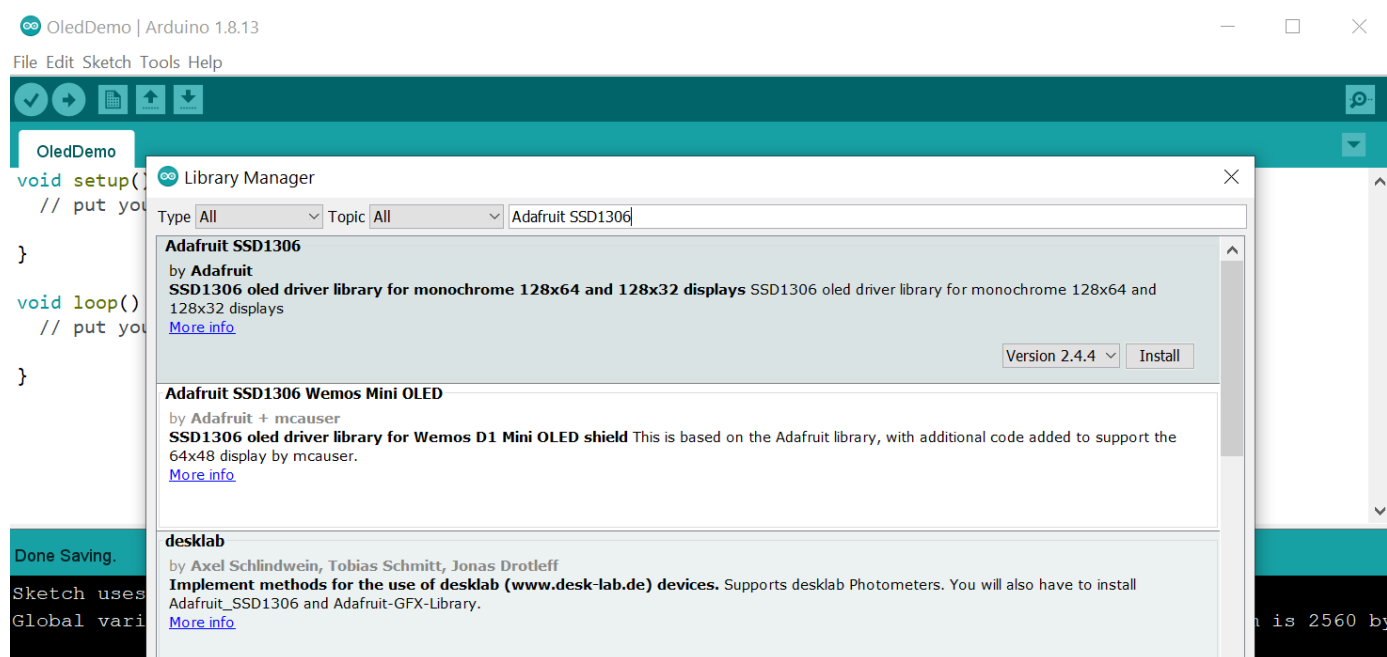


Рисунок 2.5 Менеджер бібліотек Arduino IDE

Arduino IDE 2.0

Застосунок Arduino IDE заснований на платформі Theia IDE і створений за допомогою Electron є вдосконаленням класичної IDE із підвищеною продуктивністю, покращеним інтерфейсом користувача та багатьма новими функціями, такими як автозавершення, вбудований налагоджувач і синхронізація ескізів із Arduino Cloud [6].



Рисунок 2.6 Менеджер бібліотек Arduino IDE

Перевірити/завантажити — скомпілюйте та завантажте код на плату Arduino.

Вибір плати та порту — тут автоматично відображаються виявлені плати Arduino разом із номером порту.

Sketchbook - тут ви знайдете всі свої ескізи, які локально зберігаються на вашому комп'ютері. Крім того, ви можете синхронізуватися з Arduino Cloud, а також отримувати свої ескізи з онлайн-середовища.

Менеджер плат — пакети Arduino від сторонніх розробників, які можна встановити.

Менеджер бібліотек - перегляд та встановлення бібліотек Arduino, створених Arduino та її спільнотою.

Debugger - тестування та налагодження програм в реальному часі.

Пошук - пошук ключових слів у коді.

Монітор послідовного порту — відкриває інструмент моніторингу послідовного порту, як нову вкладку на консолі.

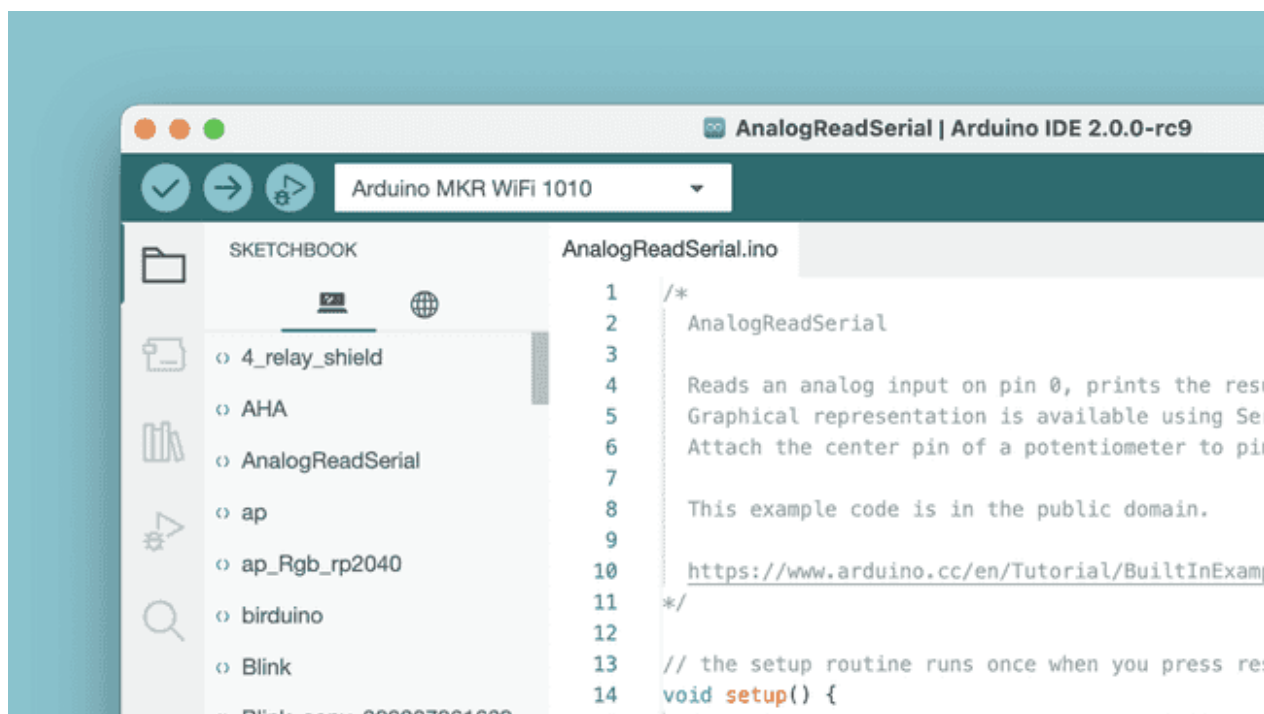


Рисунок 2.7 Скетч Arduino IDE

Скетчбук – це місце, де зберігаються файли коду. Ескізи Arduino зберігаються як .ino файли та повинні зберігатися в папці з точною назвою. Наприклад, ескіз my_sketch.ino повинні зберігатися в папці під назвою my_sketch.

За допомогою *менеджера бібліотек* ви можете переглядати та встановлювати тисячі бібліотек. Бібліотеки є розширеннями Arduino API і полегшують, наприклад, керування серводвигуном, читання певних датчиків або використання модуля Wi-Fi.

Serial Monitor — це інструмент, який дозволяє переглядати потокові дані з вашої плати, наприклад, за допомогою команди Serial.print().

Історично цей інструмент розташовувався в окремому вікні, але тепер він інтегрований із редактором. Це спрощує одночасну роботу кількох екземплярів на комп'ютері.

У новій версії плотер серійного порту отримав оновлення. Він виглядає більшим, кращим. Може відображати декілька змінних одночасно.

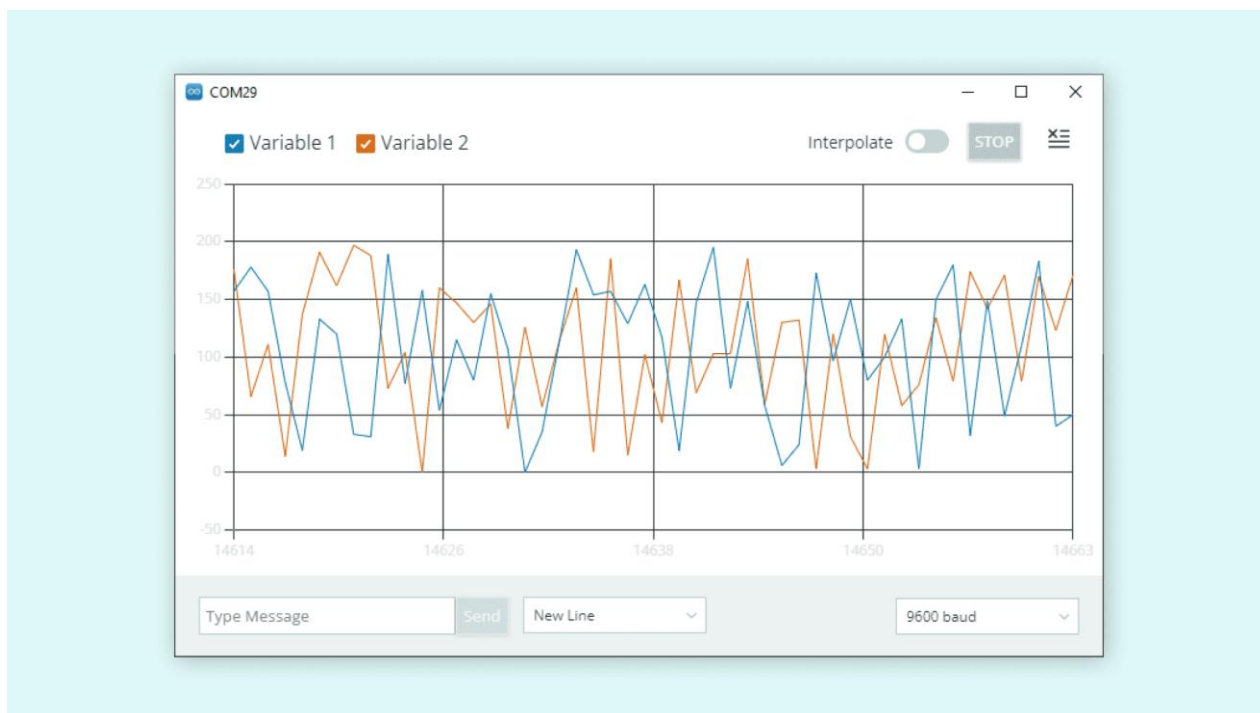


Рисунок 2.8 Serial Monitor

Також пришвидшився час компіляції та оновлення програм. Інтерфейс редактора став більш лаконічним. На бічну панель, для швидшого доступу були переміщені менеджер бібліотек, менеджер файлів, налагодження та інші. Редактор має різні кольорові теми на вибір.

Окремо варто згадати про покращене автозавершення. Це нововведення може значно підвищує продуктивність. При введенні тексту у редакторі, Arduino IDE 2.0 відображає список вибраних ключових слів, які ви можете написати. Цей список містить імена змінних і функції, знайдені у вашому вихідному коді або будь-яких завантажених бібліотеках. У старій IDE не було такої функції, але в більшості сучасних IDE є, тому ця функція була дуже затребувана більш досвідченими користувачами.

Arduino Cloud

Arduino Cloud — це платформа для розробки проектів Arduino та підключення їх до мережі інтернет. Він підтримує безпечні з'єднання з платами через Wi-Fi, LoRa, Ethernet і стільниковий зв'язок (GSM/NB-IoT), а також дозволяє створити систему для надсилання будь-якої змінної інформації, яку ви

можете придумати, з однієї плати на іншу протягом декількох хвилин після їх запуску.



Рисунок 2.9 Панель віджетів Arduino Cloud

Однією з ключових переваг Arduino Cloud є простота використання. Навіть ті, хто не має досвіду програмування, можуть швидко та легко підключити свої пристрої до хмари та почати керувати ними віддалено. Крім того, Arduino Cloud надає широкий спектр інструментів і ресурсів, які спрощують моніторинг і керування вашими пристроями, гарантуючи, що вони завжди функціонують на оптимальному рівні. Завдяки зручному інтерфейсу та потужним функціям Arduino Cloud швидко стає популярним вибором серед ентузіастів і професіоналів IoT.

Хмарна платформа Arduino складається з:

- Інтегрованого середовища розробки (IDE) для програмування плат,
- хмарний серверний сервіс для синхронізації даних із плат Arduino, клієнтів Python і JavaScript ,

- графічна панель (дашборд) для контролю та моніторингу плат та мобільний додаток.

- REST API та інструменти командного рядка для більшої автоматизації.

За допомогою Arduino Cloud можна:

1. Створити скетч для Arduino
2. Завантажити програму на свою плату через Wi-Fi.
3. Створити інформаційну панель із набором віджетів для контролю та моніторингу даних.

Arduino Cloud підтримує роботу як з власними, так і сторонніми пристроями, які можна програмувати за допомогою Python, MicroPython, Javascript або Node-RED.

Веб-редактор Arduino Cloud пропонує ту саму функціональність, що й класична Arduino IDE. Однак веб-версія дозволяє вам працювати над програмами Arduino на різних комп'ютерах без копіювання файлів або використання інструменту керування конфігурацією програмного забезпечення (SCM), наприклад GIT. Web IDE дозволяє редагувати ескізи, імпортувати бібліотеки, отримувати доступ до послідовного монітора, а також компілювати та завантажувати програми на ваші плати розробки.

Інструмент Manager for Linux дозволяє налаштовувати та керувати пристроями Linux, такими як Raspberry Pi, для використання з Arduino Cloud.

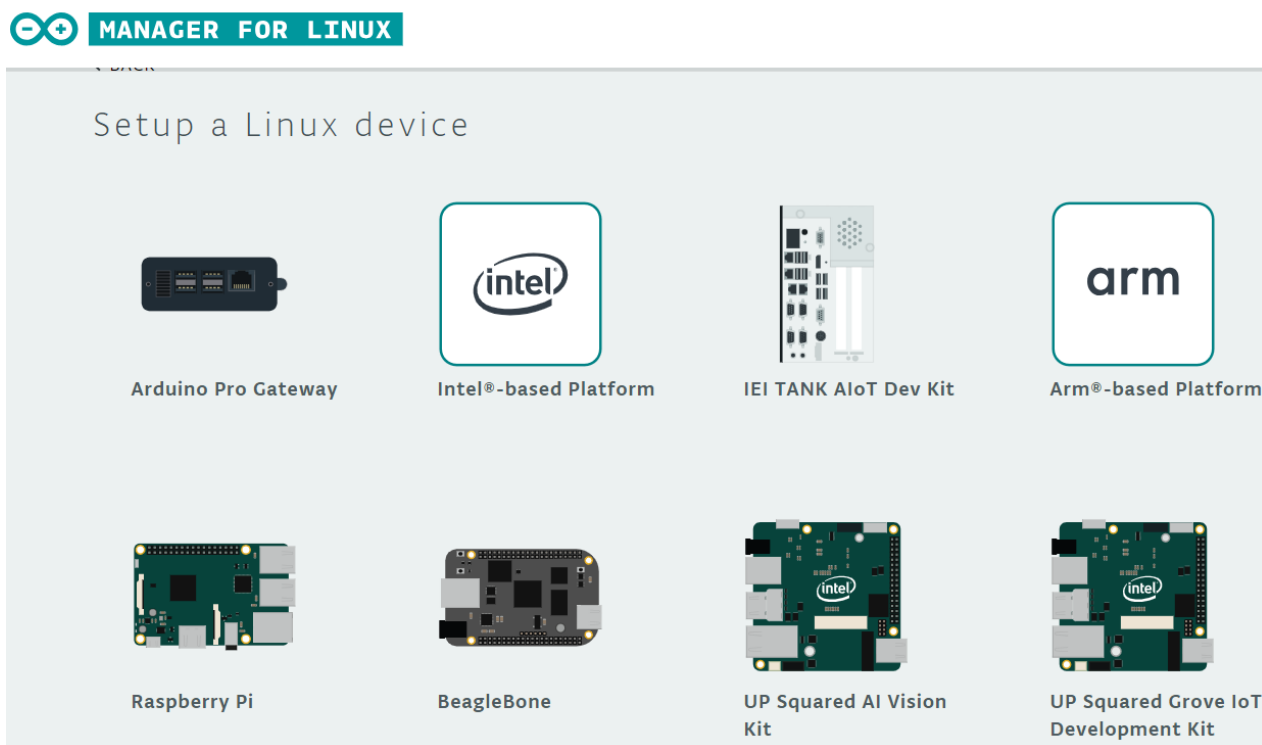


Рисунок 2.10 Інструмент Manager for Linux

Однією з головних переваг використання Arduino Cloud є можливість віддаленого доступу до ваших пристроїв і проектів. Це означає, що ви можете контролювати та керувати своїми пристроями з будь-якої точки світу, якщо у вас є підключення до Інтернету. Це особливо корисно для проектів Інтернету речей, де потрібно відстежувати та керувати пристроями в різних місцях.

Arduino Cloud також забезпечує безпечне та надійне рішення для зберігання даних для ваших проектів. Це означає, що ви можете зберігати дані зі своїх пристроїв у хмарі та мати до них доступ з будь-якої точки світу. Це особливо корисно для проектів, які створюють великі обсяги даних, оскільки ви можете зберігати та аналізувати дані без необхідності локального зберігання.

Arduino Cloud також полегшує співпрацю з іншими розробниками та інженерами над вашими проектами. Ви можете ділитися своїми проектами з іншими користувачами та працювати разом над розробкою та вдосконаленням своїх проектів. Це особливо корисно для проектів з відкритим кодом, де співпраця є ключем до успіху.

Додаток для інструментів ML на базі EdgeImpulse — це потужне, але просте у використанні рішення для машинного навчання, яке допомагає використовувати потужність машинного навчання для вилучення інформації з даних Arduino. Платформа дає змогу використовувати ці дані для створення прогностичних моделей лише за допомогою кількох рядків коду. Це бездоганно працює на мобільних пристроях і повністю інтегрований із платами Portenta H7, Nicla Vision, Nicla Sense ME, Nano33 BLE Sense та інших.

Edge Impulse — це провідна платформа для розробки машинного навчання на периферійних пристроях, безкоштовна для розробників і користується довірою компаній у всьому світі. Компанія має на меті надати кожному розробнику та виробнику пристроїв найкращий досвід розробки та розгортання машинного навчання на межі, зосереджуючись на сенсорних, аудіо- та програмах комп'ютерного зору. Завдяки Edge Impulse розробники програмного забезпечення, інженери та експерти в галузі можуть вирішувати реальні проблеми за допомогою машинного навчання на периферійних пристроях без докторського ступеня, або передові вбудовані інженерні навички. Від початку роботи до MLOps у виробництві, Edge Impulse забезпечує максимальну ефективність і швидкість на широкому спектрі обладнання від мікроконтролерів до центральних процесорів.

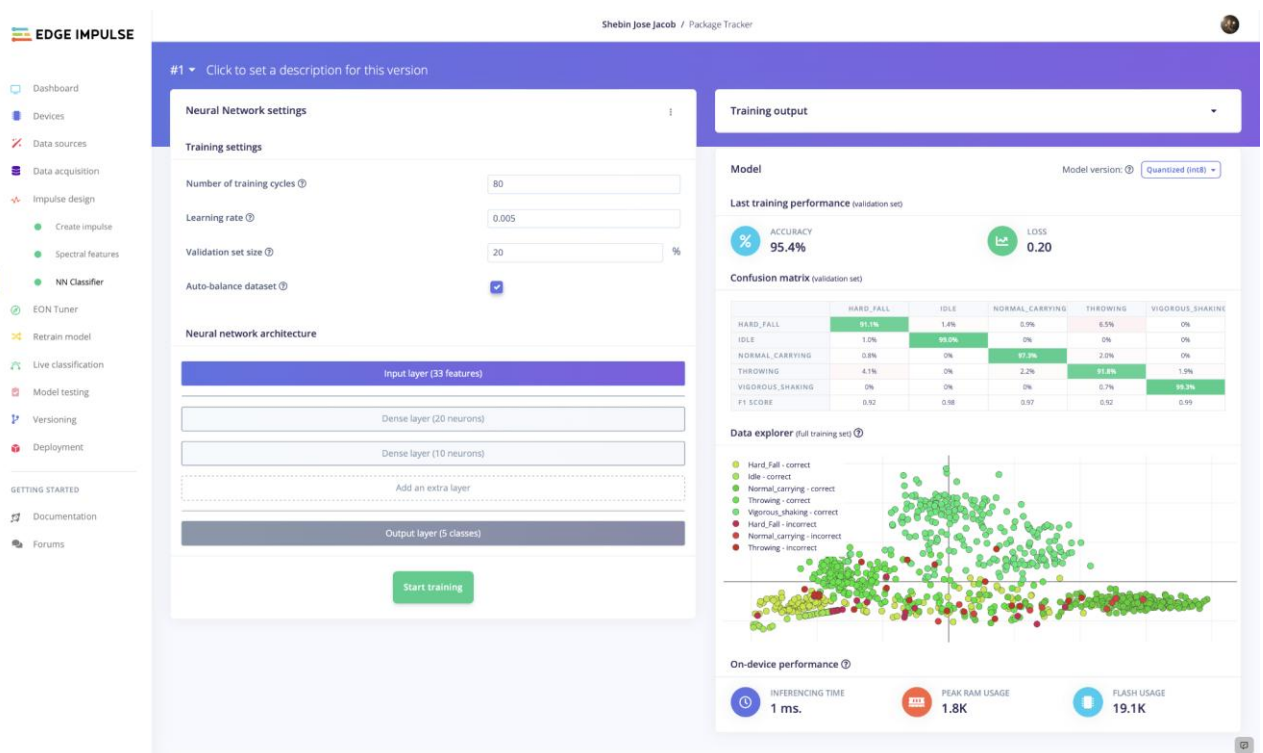


Рисунок 2.11 Інтерфейс ML платформі Edge Impulse

Запуск моделей машинного навчання (ML) на мікроконтролерах є одним із найцікавіших розробок останніх років, що дозволяє невеликим пристроям із живленням від батареї виявляти складні рухи, розпізнавати звуки або знаходити аномалії в даних датчиків.

Тенденцію до запуску ML на мікроконтролерах іноді називають Embedded ML або Tiny ML. TinyML має потенціал для створення невеликих пристроїв, які можуть приймати розумні рішення без необхідності надсилати дані в хмару – чудово з точки зору ефективності та конфіденційності. Навіть потужні моделі глибокого навчання (на основі штучних нейронних мереж) тепер досягають мікроконтролерів. За останній рік було досягнуто значних успіхів у тому, щоб зробити моделі глибокого навчання меншими, швидшими та придатними для роботи на вбудованому обладнанні завдяки таким проектам, як TensorFlow Lite for Microcontrollers, uTensor і CMSIS-NN від Arm; але створення якісного набору даних, вилучення правильних функцій, навчання та розгортання цих моделей все ще може бути складним.

2.1 Arduino PLC IDE

Arduino PLC IDE — це інтегроване середовище розробки (IDE), яке дозволяє користувачам програмувати та керувати програмованими логічними контролерами (PLC) за допомогою платформи Arduino [7]. Програмне забезпечення працює з платформами Arduino Opta та Portenta Machine Control і дозволяє програмувати ПЛК за допомогою мов стандарту Міжнародної електротехнічної комісії (IEC) 61131-3: сходова діаграма, функціональна блок-схема, структурований текст, послідовна функціональна схема та Список інструкцій.

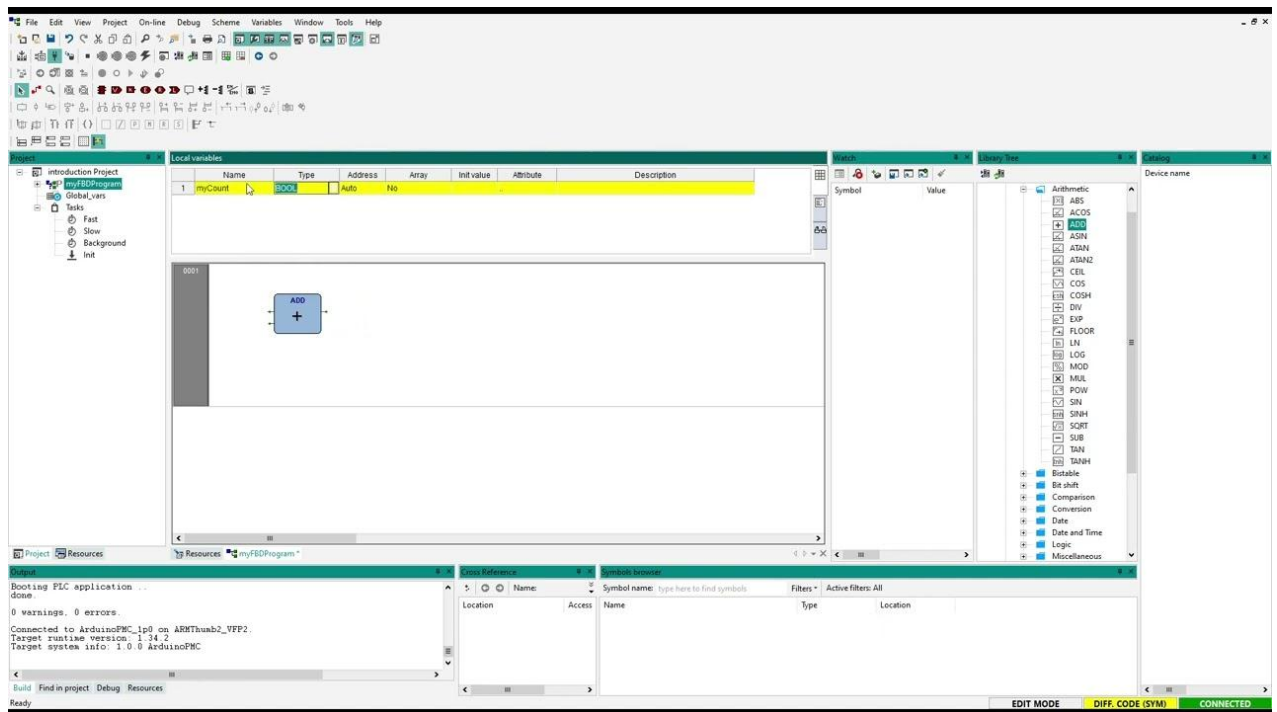


Рисунок 2.12 Інтерфейс Arduino PLC IDE

IDE підтримує п'ять стандартних мов програмування PLC, визначених IEC 61131-3:

- Ladder Diagram (LD)
- Function Block Diagram (FBD)
- Structured Text (ST)

- Instruction List (IL)
- Sequential Function Chart (SFC)

Програмне забезпечення Arduino PLC IDE розроблено для спрощення програмування ПЛК і створення проектів автоматизації. Програмне забезпечення забезпечує простий, зручний інтерфейс, який дозволяє легко налаштувати плати Arduino PLC і написати код для них. Програмне забезпечення має різні функції, які роблять його придатним для різних проектів автоматизації.

Програмне забезпечення є відкритим і безкоштовним для використання. Воно базується на платформі Arduino, популярній платформі для розробки вбудованих систем і проектів автоматизації, і програмне забезпечення розроблене для роботи з сімейством мікроПЛК Arduino Portenta та Arduino Opta.

Програмне забезпечення Arduino PLC IDE має багато функцій, які роблять його придатним для різних проектів автоматизації.

Середовище програмування: Програмне забезпечення забезпечує середовище програмування, яке дозволяє користувачам писати, редагувати та налагоджувати код для плат Arduino PLC.

Підтримка бібліотек: програмне забезпечення має різні бібліотеки, які спрощують програмування плат Arduino PLC. Бібліотеки надають ряд функцій, які можна використовувати для керування датчиками, приводами та іншими компонентами.

Інструменти налагодження: він постачається з набором інструментів налагодження, які полегшують ідентифікацію та виправлення помилок у коді, включаючи послідовний монітор, який дозволяє користувачам переглядати вихідні дані коду та виявляти помилки.

Інструменти конфігурації: користувачі, ймовірно, захочуть дізнатися більше про різні інструменти конфігурації, які спрощують налаштування плат Arduino PLC. Інструмент дозволяє користувачам налаштовувати контакти, таймери та інші параметри плати.

Управління проектами: також існують інструменти, які спрощують керування й організацію проектів автоматизації. Інструменти дозволяють користувачам створювати, відкривати та зберігати проекти.

Підтримка спільноти: програмне забезпечення підтримується великою спільнотою розробників і користувачів, які надають підтримку, керівництво та поради користувачам програмного забезпечення.

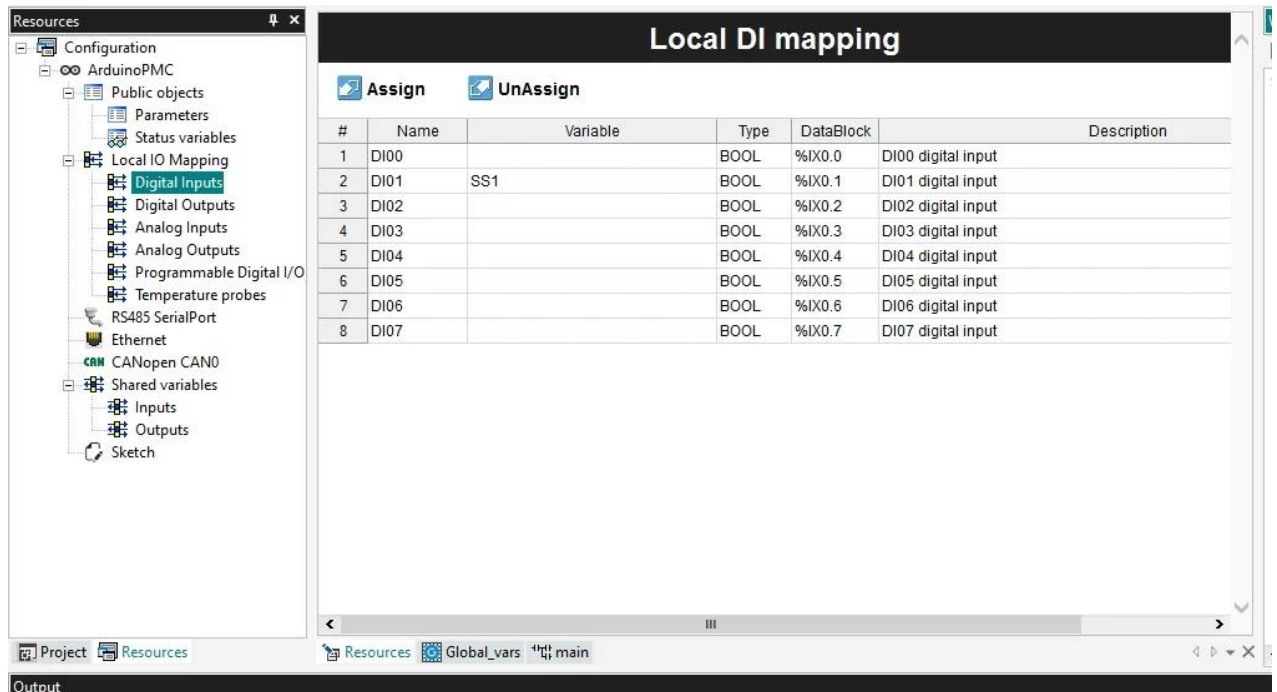


Рисунок 2.13 Меню відображення цифрових входів

Однією з ключових особливостей програмного забезпечення Arduino PLC IDE є його підтримка Modbus, який є комунікаційним протоколом, який використовується в промисловій автоматизації та дозволяє пристроям спілкуватися з іншими машинами через мережу. Програмне забезпечення Arduino PLC IDE надає бібліотеки, які спрощують реалізацію зв'язку Modbus на інших сумісних промислових платах контролерів.

Іншою важливою особливістю програмного забезпечення є його підтримка Ethernet, Bluetooth і Wi-Fi. Плата Arduino Industrial Ethernet, Portenta та Opta PLC забезпечують порт Ethernet, який можна використовувати для підключення плати до мережі. Arduino Opta Wi-Fi дозволяє підключатися до бездротової мережі.

Крім того, Arduino PLC IDE є потужним інструментом, який можна використовувати для створення складних програм PLC. Він пропонує широкий спектр функцій, зокрема:

- Підтримка стандартних мов IEC 61131-3: сходові діаграми, функціональні блок-схеми, структурований текст, послідовна функціональна схема та список інструкцій
- Підтримка кількох плат, таких як Arduino Portenta та Arduino Opta
- Підтримка кількох протоколів зв'язку, таких як Modbus і Modbus TCP
- Підтримка кількох інструментів налагодження: вікна спостереження та точки зупинки з покроковим виконанням, тригери, осцилограф і режим налагодження в реальному часі

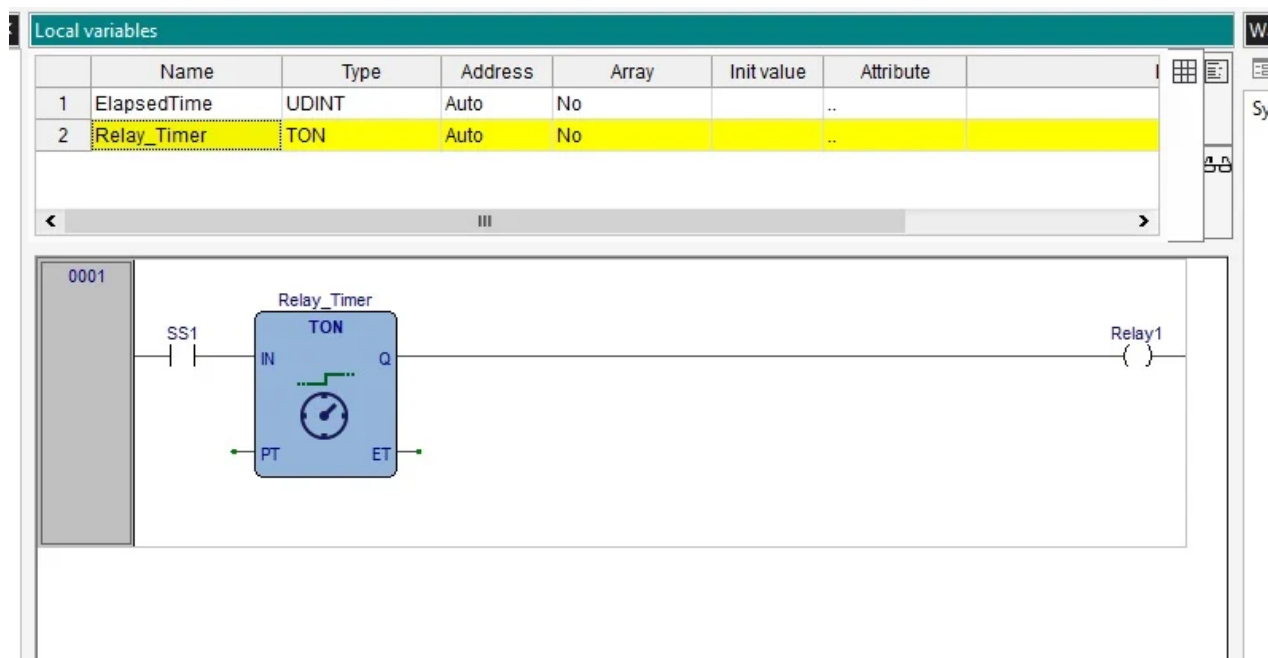


Рисунок 2.14 Інтерфейс Arduino PLC IDE

Arduino PLC IDE — це промислове програмне забезпечення для програмування та керування платформами керування Opta PLC і Portenta Machine. Програмне забезпечення надає низку функцій, включаючи підтримку бібліотек, інструменти налагодження, налаштування та керування проектами. Ці інструменти та підтримка спільноти спрощують створення проектів автоматизації

Arduino PLC IDE для широкого спектру промислових, розумних домашніх та інших програм.

Завдяки підтримці зв'язку Modbus, Ethernet і Wi-Fi, програмне забезпечення Arduino PLC IDE є універсальним і гнучким рішенням, яке використовується в промислових і інтелектуальних проектах домашньої автоматизації та роботах.

2.2 Дротові протоколи комунікації

Інтернет речей (ІоТ) - це концепція, що перетворює наше повсякденне життя, впроваджуючи підключені до Інтернету пристрої для збору, обміну та аналізу даних з метою поліпшення нашого життя та робочого середовища. Однак для забезпечення цієї неймовірної спрямованості на інтеграцію та обмін даними між пристроями, необхідні ефективні проводові протоколи.

Ці проводові протоколи використовуються для забезпечення надійного та швидкого обміну інформацією між ІоТ-пристроями, забезпечуючи таким чином роботу інтелектуальних систем, які включають в себе різноманітні пристрої від розумних термостатів та освітлення до індустриальних сенсорів та контролерів.

Серед найважливіших проводових протоколів в контексті ІоТ варто відзначити USB, UART/USART, RS-232, RS-485, Ethernet, I2C та SPI. Кожен з цих протоколів має свої унікальні характеристики, що роблять їх придатними для конкретних завдань та умов експлуатації. З їхньою допомогою забезпечується не тільки ефективний обмін даними, а й оптимізація роботи всієї мережі підключених пристроїв, що стає важливим кроком у розвитку інтерконектованого світу:

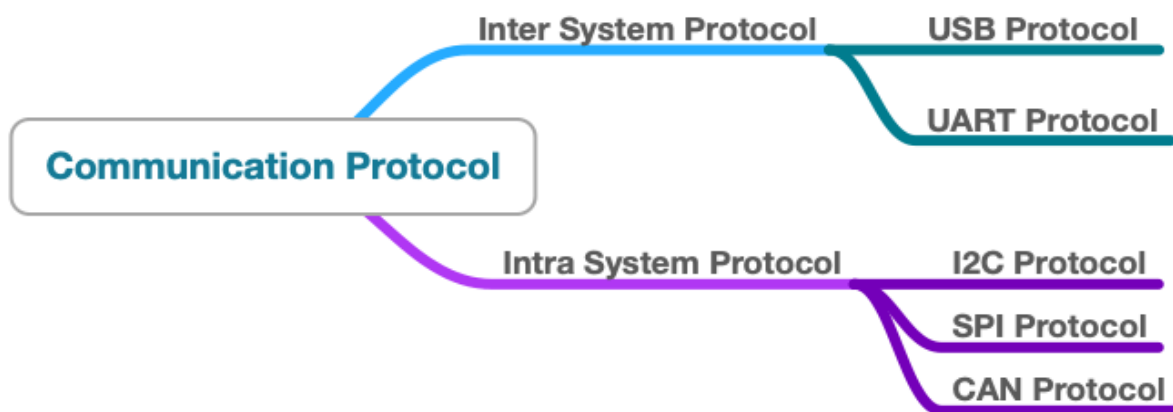


Рисунок 2.15 Дротові протоколи

USB - це серійний протокол для підключення різних пристроїв через USB-порти, такі як ігрові консолі, мобільні телефони та інші. Він забезпечує швидкий майстер-раб інтерфейс і підтримує до 127 пристроїв. Його переваги - простота та швидкість, але вимагає потужного головного пристрою та встановлення драйверів.

UART та *USART* - це протоколи для асинхронної чи синхронної передачі даних між пристроями. Вони використовують всього два проводи і не вимагають годинникового сигналу. *UART* застосовується у вбудованих системах та комп'ютерах. Його переваги - відсутність необхідності в годиннику, але є обмеження на розмір кадру та неможливість багатомайстерного з'єднання.

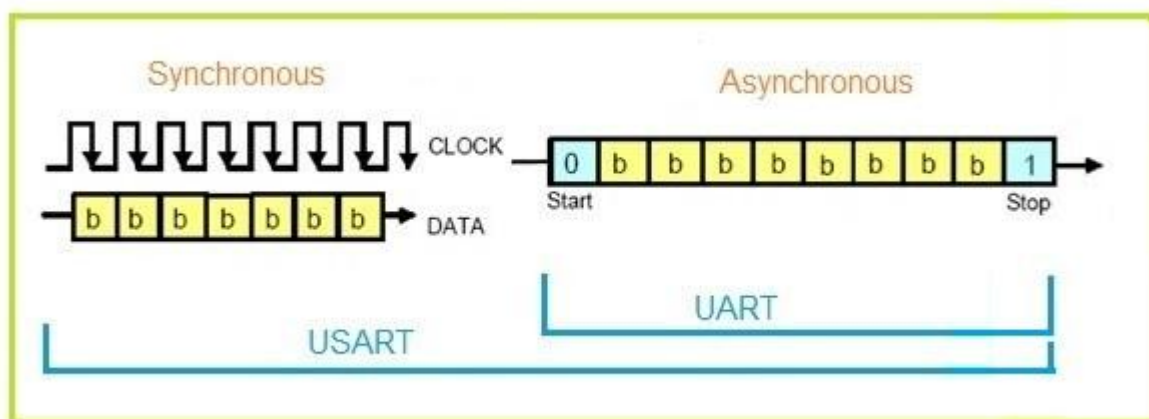


Рисунок 2.16 Схема роботи UART протоколу

RS-232 - це стандарт інтерфейсу для послідовної передачі даних через порти комп'ютера. Він є дешевим та легко доступним, але працює в режимі півдуплексного майстра-раба та має менш стандартизовані роз'єми.

RS-485 - покращена версія *RS-232*, придатна для з'єднання 10-32 пристроїв одночасно з доброю стійкістю до шумів та можливістю багатокрапкового використання [19]. Однак, він також працює в режимі півдуплексного майстра-раба та використовує менш стандартизовані роз'єми.

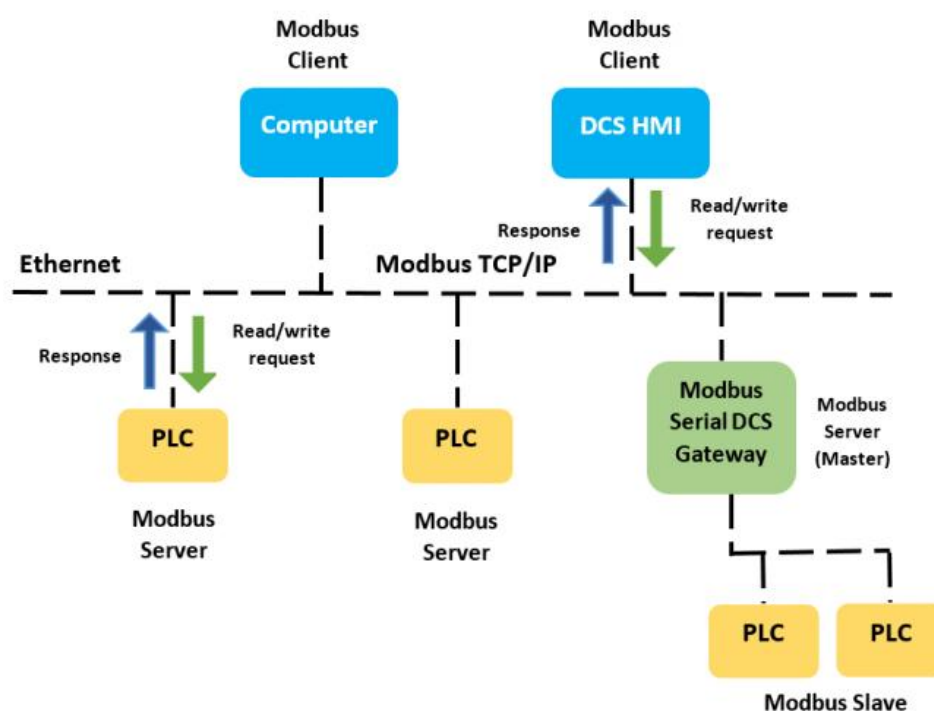


Рисунок 2.17 Принцип роботи протоколу 1 Wire

Ethernet - це мережа, яка об'єднує пристрої для передачі даних у локальній мережі (LAN). Її переваги - управління мережею за допомогою переключника, але не підходить для великих відстаней та вимагає багато проводів.

I2C - це внутрішній протокол для обміну даними між пристроями на одному колі. Він дозволяє підключати багато пристроїв без додаткових проводів, але вимагає складного обладнання та програмування.

SPI - це синхронний протокол для короткодістанційного зв'язку за допомогою чотирьох проводів. Він швидший і може об'єднувати багато пристроїв,

але вимагає більше проводів та обмежує можливість спілкування між пристроями [21].

I Wire - це протокол для простої одножильної мережі, що забезпечує обмін даними з мінімальним використанням проводів.

DALI - це цифровий протокол для управління та контролю світловими пристроями, забезпечуючи індивідуальне керування світловими точками через цифровий сигнал [22].

KNX - це стандарт для автоматизації будівель та дому, дозволяючи інтеграцію інтелектуальних пристроїв для управління освітленням, опаленням та іншими системами у різних приміщеннях.

3 СИСТЕМА ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ З PLC ARDUINO OPTA

Системи домашньої автоматизації дозволяють дистанційно керувати електронними пристроями та виконувати автоматичні процедури на основі програмних уподобань користувача, роблячи домашнє середовище більш зручним, комфортним та ефективним, а також оптимізуючи витрати на електроенергію.

[15] Програми домашньої автоматизації дозволяють, наприклад, інтелектуальне керування освітленням і контроль вентиляції та кондиціонування повітря. Завдяки можливостям промислового Інтернету речей Opta є чудовим продуктом для додатків домашньої автоматизації.



Рисунок 3.1 Opta з підтримкою підключення Wi-Fi

На наступній схемі показані електричні з'єднання передбачуваного застосування:

беруть участь у мережі, протягом кількох мілісекунд всесвітнього координованого часу (UTC).

Після синхронізації RTC Opta необхідно запрограмувати відкриття та закриття ролетних штор за допомогою двигуна контролера в певні години щодня:

Штора автоматично відкривається вранці, коли сходить сонце (наприклад, о 6:00) і закривається ввечері (наприклад, о 18:00).

Кінцеві вимикачі NO SPDT забезпечать Opta зворотний зв'язок для верхньої або нижньої кінцевої межі сонцезахисного екрану, викликаючи зупинку двигуна

Приклад ескізу застосунку

Далі пояснюється ескіз. Перша частина ескізу містить деякі визначення, необхідні для:

- Підключення Wi-Fi до мережі

- Налаштування конфігурації клієнта NTP

Оголошення змінних, пов'язаних з інтервалом оновлення RTC і запрограмованими сценаріями для керування шторами. Для підключення до Wi-Fi та сервера часу використовуються бібліотеки WiFi та NTPClient. Також використовується метод керування часом від Mbed.

```
// Бібліотеки, використані в скетчі
#include <WiFi.h>
#include <NTPClient.h>
#include <mbed_mktime.h>

// Налаштування Wi-Fi
char ssid[] = "YOUR_WIFI_SSID";
char pass[] = "YOUR_WIFI_PASSWORD";
int status = WL_IDLE_STATUS;

// Налаштування часу оновлення NTP клієнта
WiFiUDP ntpUDP;
```

```

NTPClient timeClient(ntpUDP, "pool.ntp.org", -6*3600, 0);
unsigned long interval = 60*30*1000UL;
unsigned long lastTime = 0;

//Налаштування часу відкриття та закриття ролетів
int local_hour;
int local_minutes;
int programmed_hour_1 = 6;
int programmed_hour_2 = 18;
int programmed_minutes_1 = 0;
int programmed_minutes_2 = 0;

```

Змінні `programmed_hour` та `programmed_minutes` налаштовують час відкриття штор і їх можна змінити під власні потреби.

Налаштування PLC Opta виглядає таким чином:

```

// Ініціалізація Opta
void setup() {
  Serial.begin(9600);
  while (!Serial) {
    ;
  }
  delay(5000);

  // Підключення до Wi-Fi
  while (status != WL_CONNECTED) {
    Serial.print("- Attempting to connect to WPA SSID: ");
    Serial.println(ssid);
    status = WiFi.begin(ssid, pass);
  }
}

```

```
    delay(500);
}

//Відображення інформації про мережу у виводі Serial Monitor
Serial.println();
Serial.println("- NETWORK INFORMATION");
Serial.print("- You're now connected to the network ");
printCurrentNet();
printWifiData();
delay(5000);

// Ініціалізація оновлення часу
timeClient.begin();
updateTime();

// Цифрові входи, цифрові виходи, ініціалізація вбудованих світлодіодів
pinMode(LED_D0, OUTPUT);
pinMode(LED_D1, OUTPUT);
pinMode(D0, OUTPUT);
pinMode(D1, OUTPUT);
pinMode(A2, INPUT);
pinMode(A3, INPUT);

// Зупинка руху двигуна штор
digitalWrite(LED_D0, LOW);
digitalWrite(LED_D1, LOW);
stop_shade();
}
```


У функції налаштування є п'ять основних кроків:

- Ініціалізація послідовного порту на видкосіт 9600 бод; він використовується у програмі для налагодження.

- Спроба підключення до Wi-Fi та
- Відображення інформації про мережу в виводі послідовного порту
- Ініціалізація NP клієнта та оновлення часу
- Запуск та зупинка (stop_shade) двигуна ролетних штор.

Так виглядає основний цикл коду:

```
void loop() {
// Перевірка напрямку руху штор

local_hour = getLocalHour();
local_minutes = getLocalMinutes();

if (programmed_hour_1 == local_hour && programmed_minutes_1 ==
local_minutes) {
    Serial.println("- Rolling down shade!");
    roll_down_shade();
} else if (programmed_hour_2 == local_hour && programmed_minutes_2 ==
local_minutes) {
    Serial.println("- Rolling up shade!");
    roll_up_shade();
} else {
    stop_shade();
}

// Періодичне оновлення часу клієнта
unsigned long currentTime = millis();
if (currentTime - lastTime >= interval) {
```

```

updateTime();
lastTime = currentTime;
}
}

```

Основний цикл можна розділити на два основні етапи:

- Функції `getLocalHour()` і `getLocalMinutes()` використовуються для отримання місцевого часу та порівняння із запрограмованим часом сценарію (6:00 ранку та 6:00 вечора).
- RTC періодично оновлюється (у прикладі кожні 30 хвилин) за допомогою функції `updateTime()`

Додавання нового пристрою до Arduino Cloud

Для підключення Arduino Opta до IoT Cloud, зареєструйтесь та перейдіть до Arduino Cloud у розділ Devices. Натисніть Add Device [11].

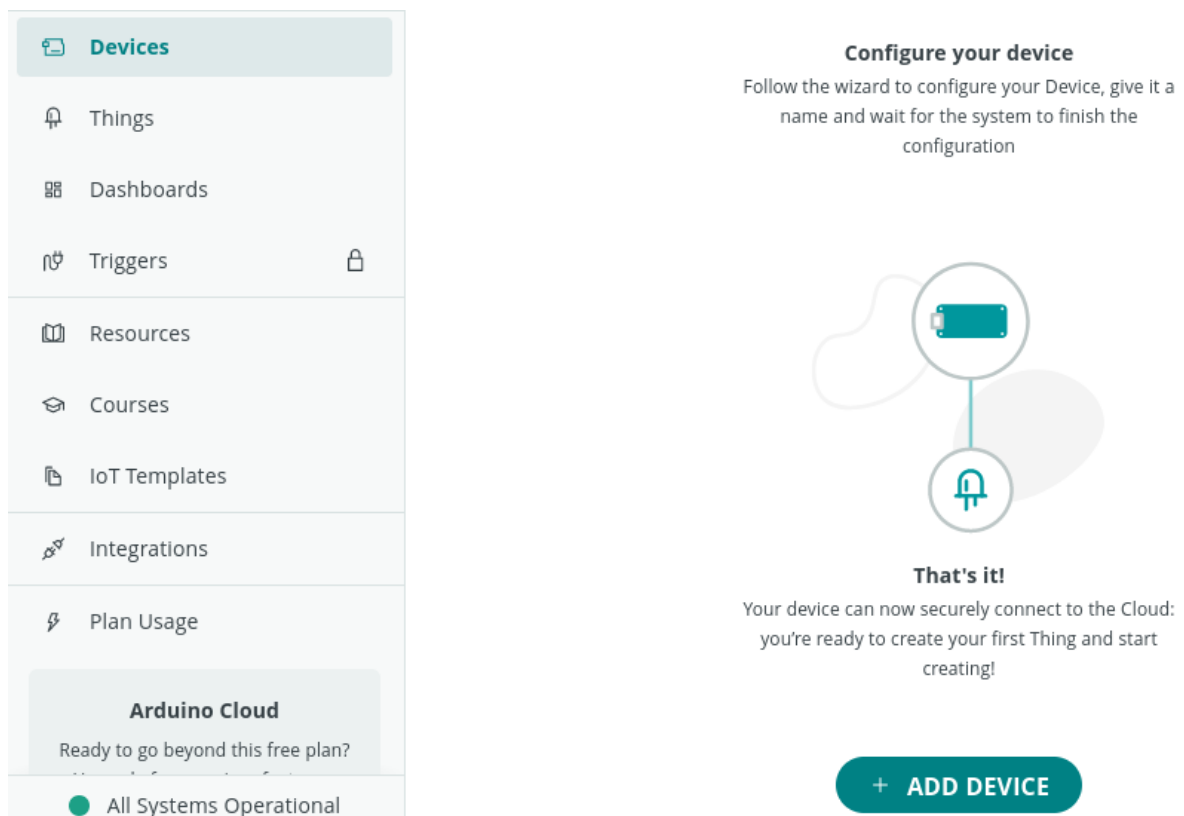


Рисунок 3.3 Додавання пристрою до Arduino Cloud

У вікні, що відкрилось оберіть автоматичний режим та пристрої Arduino

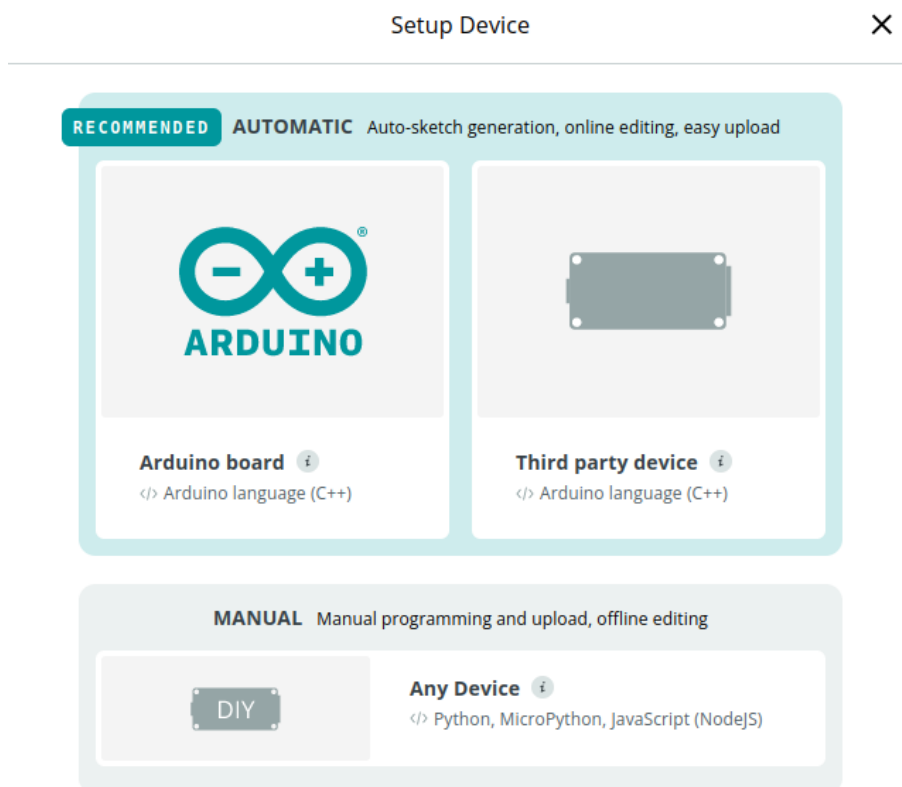


Рисунок 3.4 Налаштування Arduino Cloud Agent
Завантажте та встановіть Arduino Cloud Agent.

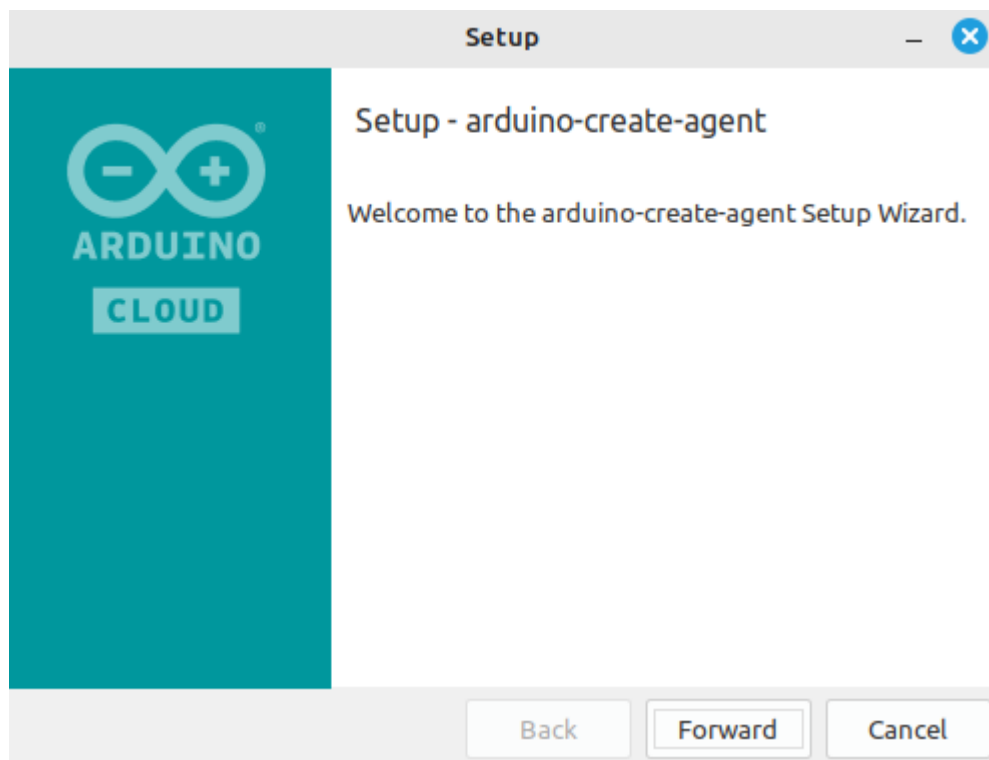


Рисунок 3.5 Завантаження Arduino Cloud Agent

Підключіть Arduino Opta до комп'ютера за допомогою USB кабелю та до інтернету за допомогою Ethernet кабелю. Після знаходження пристрою з'явиться вікно знаходження пристрою.

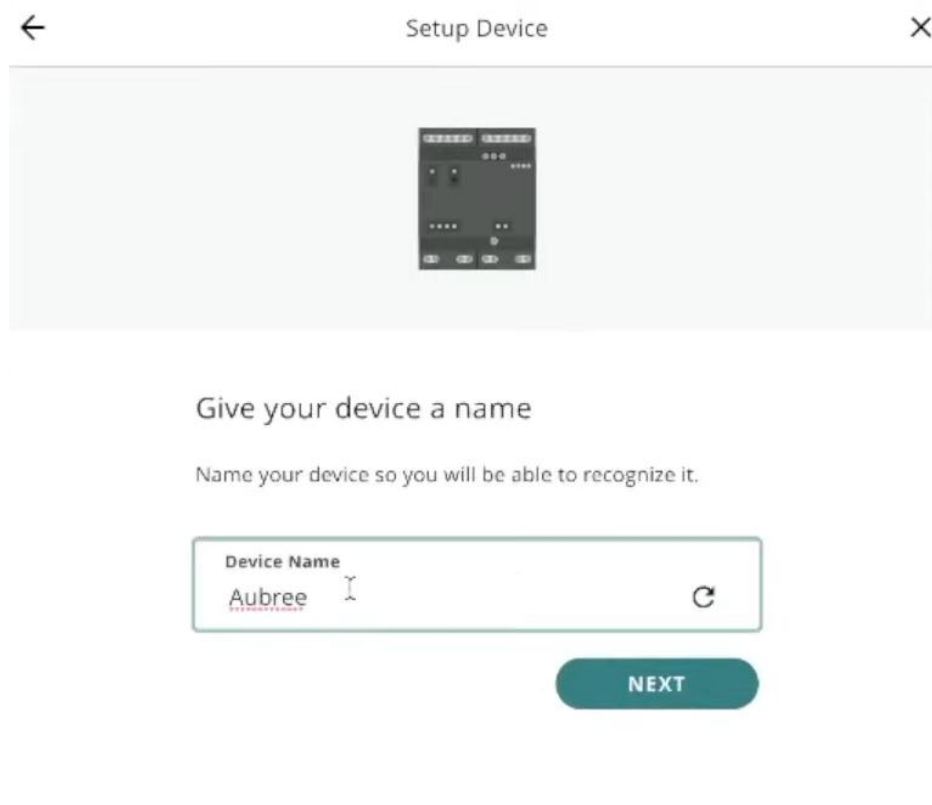


Рисунок 3.6 Підключення Arduino Opta до Arduino Cloud

Натисніть кнопку налаштувати, дайте своєму пристрою назву та виберіть тип мережевого підключення.

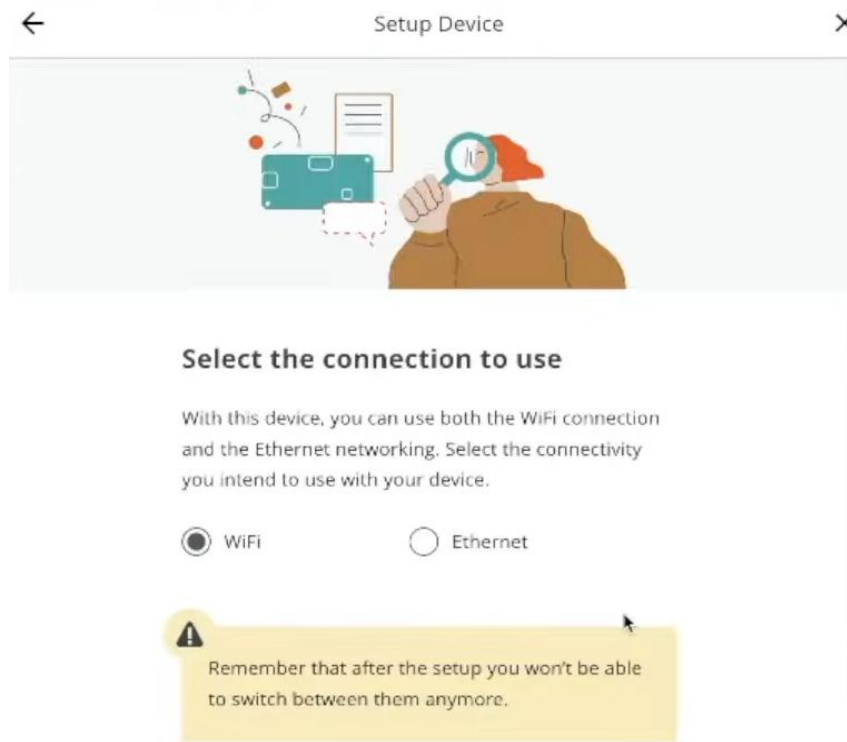


Рисунок 3.7 Налаштування Arduino Cloud

У цьому прикладі ми будемо використовувати підключення Wi-Fi; Проте також можна використовувати Ethernet. Пристрій налаштуватиметься певний час, а після видасть ID

Інтеграція скетчу Opta з Arduino Cloud

Пояснений вище приклад коду можна змінити так, щоб керувати контролером штор, через Arduino Cloud [24]. Для цього до коду додамо розширені функції, такі як віддалене програмування, активація та моніторинг:

Зміна часу відкриття та закриття.

Відкриття та закриття ролетної штори в будь-який момент.

Контроль стану ролетних штор (відкрито чи закрито).

Після пов'язування Opta з обліковим записом Arduino IoT Cloud [28] створіть речі (things) з змінними описаними нижче:

змінна: `programmed_hour_1`, тип: `int`, дозвіл: `Read & Write`, Політика оновлення: `On change`, Опис: ця змінна визначає ранкову годину відкриття ролет.

змінна: `programmed_hour_2`, тип: `int`, дозвіл: `Read & Write`, Політика оновлення: `On change` Опис: ця змінна визначає вечірню закриття ролет.

За аналогією робимо наступні змінні

`programmed_minutes_1`: `int` (`Read & Write`, `On change`)

Опис: Ця змінна визначає хвилини першого запрограмованого сценарію.

`programmed_minutes_2`: `int` (`Read & Write`, `On change`)

Опис: Ця змінна визначає хвилини другого запрограмованого сценарію.

`open_shade`: `bool` (`Read & Write`, `On change`)

Опис: Ця змінна використовується для відкриття ролетної штори, якщо її встановлено `true`.

`close_shade`: `bool` (`Read & Write`, `On change`)

Опис: Ця змінна використовується для закриття ролетної штори, якщо для неї встановлено значення `true`.

`shade_shade`: `bool` (`Read & Write`, `Periodically`)

Опис: Ця змінна використовується для моніторингу поточного стану ролетної штори, `true` для відкритих і `false` для закритих.

Add variable ×

Name
programmed_hour_1

↻ Sync with other Things ?

Integer Number eg. 1 ▾

Declaration
`int programmed_hour_1;` ?

Variable Permission ?

Read & Write
 Read Only

CANCEL ADD VARIABLE

Рисунок 3.8 Приклад створення змінної в Arduino Cloud

З змінних створимо інформаційну панель і віджети:

∞
IOT CLOUD

Things
Dashboards
Devices
Integrations
Templates

UPGRADE PLAN

👁
✎

📱
🖱

Blind Controller Dashboard

Programmed hour 1

19

Programmed minutes 1

5

Programmed hour 2

19

Programmed minutes 2

30

✓

Status

Open shade

Close shade

Рисунок 3.9 Керування двигуном ролетних штор через Arduino Cloud

3.1 Управління енергією за допомогою Opta

PLC Opta може стати незамінною підтримкою для домашнього енергоменеджменту [14]. Отримання інформації про споживання електроенергії та надання статистики використання з сезонними прогнозами можуть допомогти в плануванні та управлінні електричними пристроями для оптимізації енергоефективності. Завжди бути на зв'язку та отримувати інформацію, зручно, використовуючи хмару Arduino IoT, і додайте можливість самоналаштування, відстежуючи та реєструючи електричну статистику, а також опцію керування підключеними пристроями на вимогу на основі попередньо встановлених тригерів.

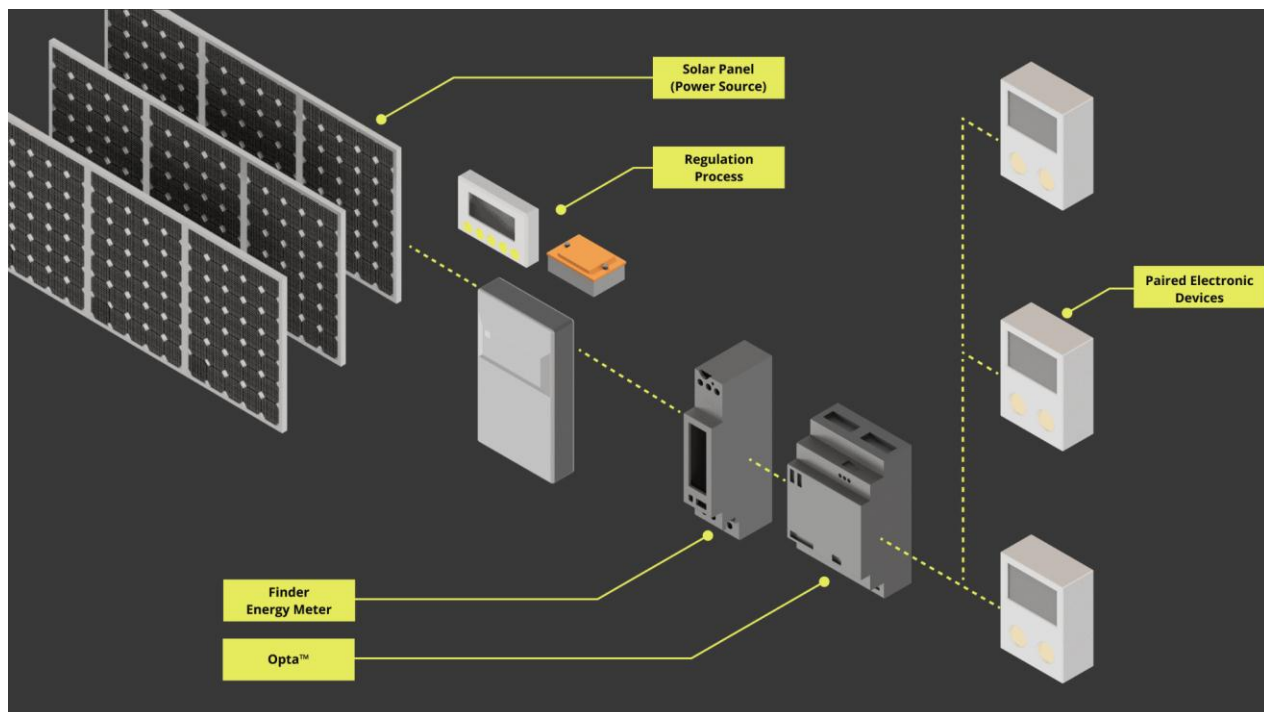


Рисунок 3.10 Приклад використання Arduino Cloud в системі керування енергією

За допомогою Opta можна отримувати та обробляти команди дистанційного керування з Arduino Cloud, увімкнути Opta для керування пристроями на основі моделей споживання енергії користувачами та енергії, доступної від сонячних панелей або будь-яких інших джерел живлення.

Автоматизувати роботу побутової техніки, враховуючи вимоги користувачів, доступність електроенергії та енергоефективність. Нижче наведено візуальне представлення передбачуваного застосування.

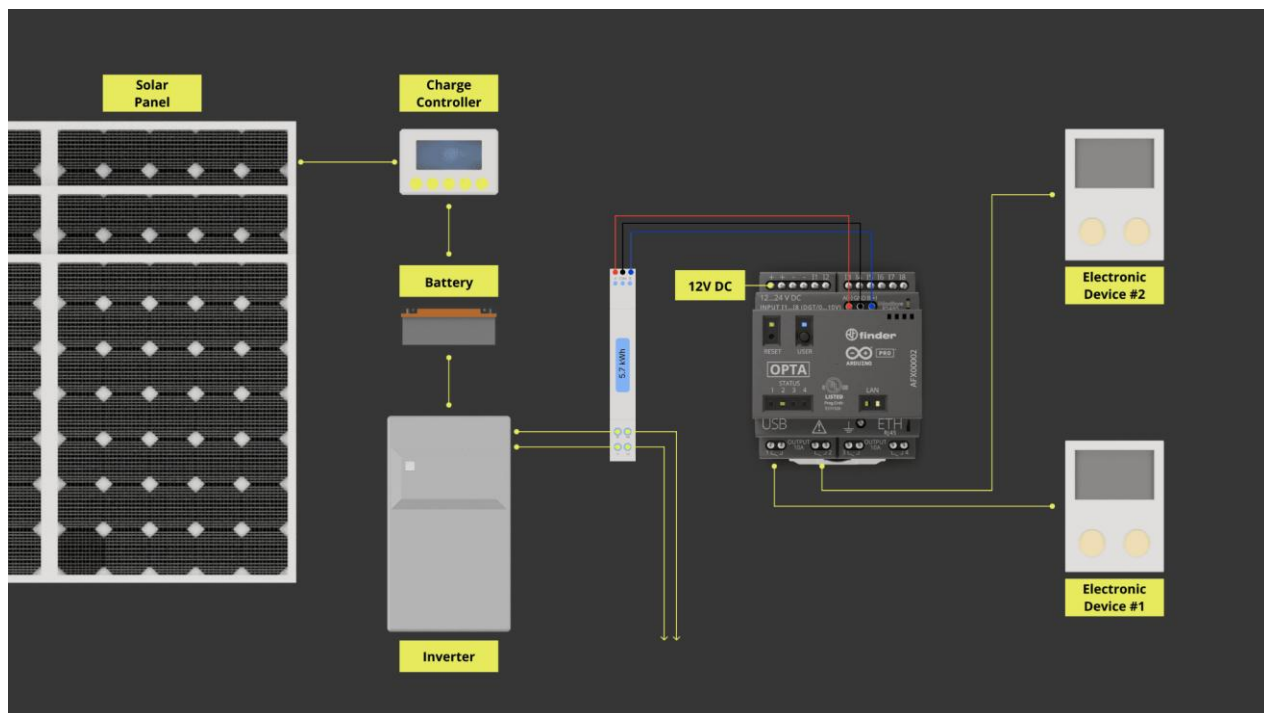


Рисунок 3.11 Приклад використання Arduino Cloud в системі керування енергією

Система Opta матиме доступ до деталей споживання в режимі реального часу з лічильника енергії, використовуючи Modbus RTU через інтерфейс RS-485. Енергія від сонячних панелей проходить кілька процесів, перш ніж досягти лічильника енергії. Побутовою технікою можна керувати за допомогою вбудованих релейних функцій системи Opta. Також варто відзначити, що сонячні батареї можуть замінити інші джерела живлення.

Основна роль Opta полягає в тому, щоб ефективно використовувати електроенергію, використовуючи дані лічильника енергії, підключеного до сонячної панелі. Він отримує та обробляє дані з лічильника енергії, оцінюючи споживання в реальному часі на основі порогових значень лічильника та поточної вихідної потужності сонячної панелі.

Для цієї програми ми використовуємо модель лічильника енергії 7M.24 від Finder. Ви можете отримати доступ до його таблиці даних тут . Ця модель спілкується через Modbus RTU через інтерфейс RS-485. Функції реле Opta™

керуватимуть відповідними побутовими приладами. Щоб зібрати дані та контролювати розподіл електроенергії, Opta™ виконує такі дії:

Отримати показання напруги та струму з лічильника електроенергії.

Зберіть три типи показників потужності з лічильника електроенергії: Загальна активна потужність - P_t (W) , Загальна реактивна потужність - Q_t (var) і повна видима потужність - S_t (VA) .

Упорядкуйте зібрані дані про напругу, струм і потужність (активну, реактивну та уявну) у різні категорії, щоб відобразити фактичне , середнє , максимальне та мінімальне значення кожного.

Отримайте доступ до цифр лічильника енергіїWh і varh одиниць.

Оптимально розподіляйте потужність для регулювання вибраних побутових приладів відповідно до енергетичного профілю користувача.

Хоча всі ці процеси обробляються Opta локально, вона також підключена до Arduino Cloud через Wi-Fi. Це підключення дозволяє користувачам переглядати споживання енергії та дистанційно керувати підключеними пристроями через Arduino Cloud.

Приклад коду Opta Energy Management

Наданий код демонструє можливості Opta, описані раніше. Варто зазначити, що деякі функції коду генеруються Arduino Cloud під час налаштування інформаційної панелі. [30]

Код вимагає включення певних заголовків. Ці заголовки дозволяють працювати з інтерфейсом RS-485, протоколом Modbus RTU, з'єднанням Arduino Cloud і планувальником. Планувальник контролює обмін даними через інтерфейс RS-485 за протоколом Modbus RTU. Крім того, він містить параметри, необхідні для стабільного зв'язку, дотримуючись стандартів Modbus RTU.

```
#include "stm32h7xx_ll_gpio.h"
#include "thingProperties.h"
#include <ArduinoModbus.h>
#include <ArduinoRS485.h>
```

```

#include <Scheduler.h>

#define F7M24 0x21
#define pause_trigger 15

float V_actual, V_avg, V_max, V_min;
float A_actual, A_avg, A_max, A_min;
float W_actual, W_avg, W_max, W_min;
float Var_actual;
float Va_actual, Va_avg, Va_max, Va_min;
float Wh_packet, Varh_packet, Wh_Abs_packet;

typedef struct uPWR_STRUCTURE {
    float uV_code;
    float uW_code;
    float uWh_code;
} PWR_STRUCTURE;
PWR_STRUCTURE user_profile;
float operation_safety_margin = 1.1;
float estimated_max_power = 220;
float estimated_max_energy = 2880;
float Device_1_Limiter = 1;
float Device_2_Limiter = 100;
float Device_1_CompRef = A_actual;
float Device_2_CompRef = W_avg;
constexpr auto baudrate { 19200 };

constexpr auto bitduration { 1.f / baudrate };
constexpr auto preDelayBR { bitduration * 9.6f * 3.5f * 1e6 };

```

```
constexpr auto postDelayBR { bitduration * 9.6f * 3.5f * 1e6 };
```

Підключення Opta до Arduino Cloud

Після налаштування коду можна створити подібну інформаційну панель Arduino Cloud.

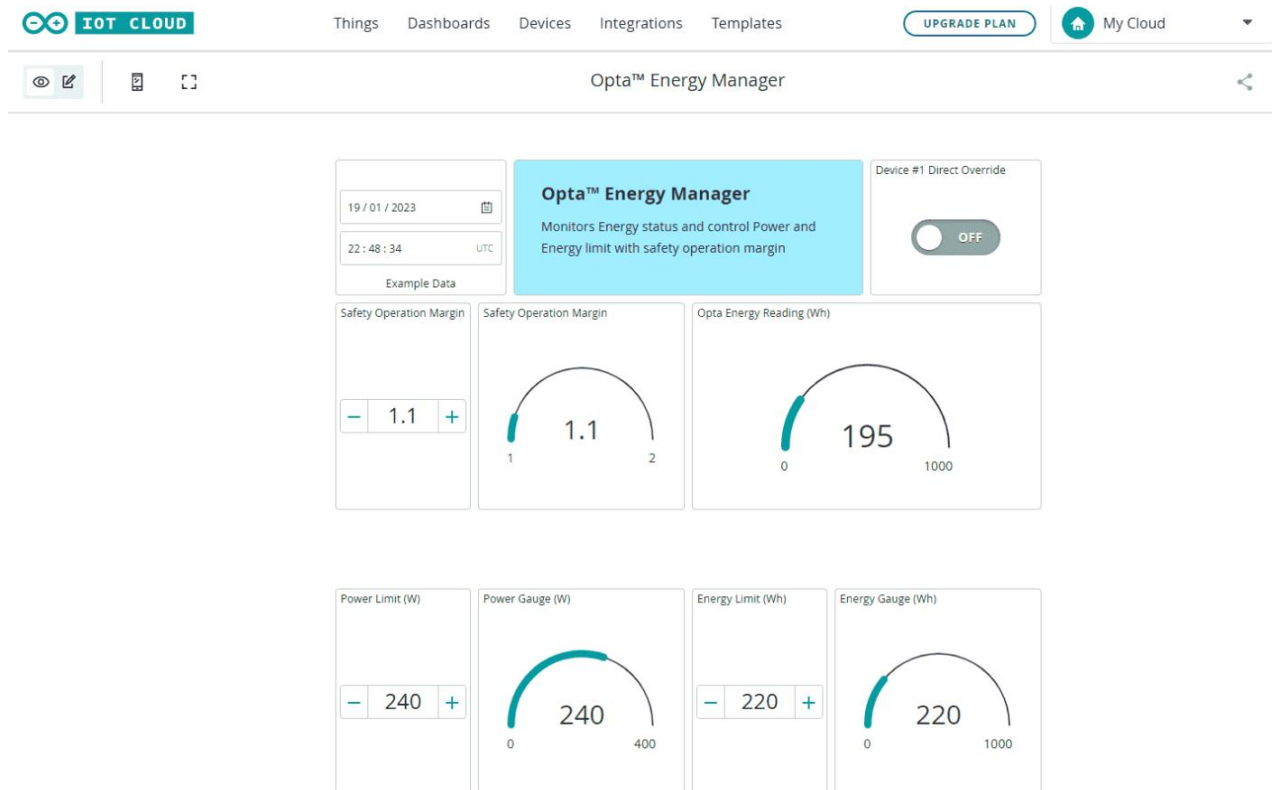


Рисунок 3.12 Керування енергоменеджментом через Arduino Cloud

3.2 Рекомендації щодо створення дротового розумного будинку на основі PLC

Вибір кабельної топології

У світі дротових розумних будинків вибір топології для створення мережі грає ключову роль у забезпеченні ефективності та стабільності системи. Один із найбільш перспективних підходів - це топологія зірка, що виявляється ефективнішою в порівнянні з послідовним підключенням пристроїв.

Важливо відзначити, що у топології зірка кожен пристрій безпосередньо підключений до центрального вузла чи контролера. Це не лише полегшує керування системою, але і робить її більш стійкою до можливих несправностей. Наприклад, якщо один пристрій виявиться несправним, це не вплине на роботу інших, завдяки ізольованому підключенню.

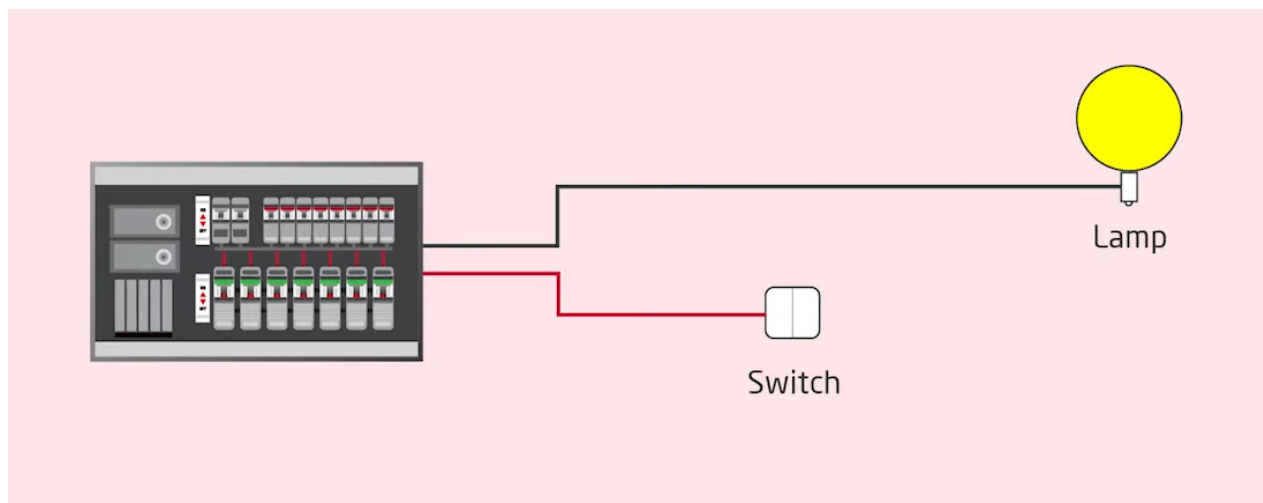


Рисунок 3.13 Топологія зірка

Крім того, топологія зірка забезпечує легкість розширення системи. Додавання нових пристроїв не вимагає перегляду всього ланцюга, як це може трапитися у послідовному підключенні. Це робить систему більш гнучкою та готовою до майбутніх розширень.

Однією з ключових переваг є також можливість легкої ідентифікації та усунення несправностей. Коли кожен пристрій має свій окремий шлях до центрального вузла, виявлення та вирішення проблем стає значно простіше.

Неабиякою перевагою топології зірка є зручне керування та моніторинг. Централізований доступ до кожного пристрою забезпечує ефективне адміністрування системи, спрощуючи контроль та налаштування. Такий підхід полегшує життя власників дротових розумних будинків, дозволяючи легко налаштовувати та контролювати всі аспекти їхнього домашнього середовища.

Визначення груп освітлення

У випадку, коли домагаємося індивідуального керування кожною лампою чи розеткою в контексті розробки дротового розумного будинку, рекомендується

розглянути можливість індивідуального подачі електроживлення для кожного зазначеного пристрою. Такий підхід забезпечить повний контроль над освітлювальними та електропостачальними вузлами, що, в свою чергу, уможливить широкий спектр налаштувань для регулювання параметрів пристроїв залежно від конкретних вимог користувача.

У випадку, коли такий індивідуальний контроль не є первинною вимогою, можливою альтернативою є агрегація пристроїв у групи зі спільним функціоналом. Це визначення груп, в яких лампи чи розетки працюватимуть синхронно. Подібний підхід дозволяє впроваджувати оптимізовані режими споживання електроенергії та ефективно керувати групами пристроїв, забезпечуючи ефективність та зручність в управлінні домашнім середовищем.

Визначення груп розеток

Якщо ми хочемо керувати кожною розеткою окремо потрібно прокласти кабель до кожної розетки окремо. Якщо необхідне керування групою, наприклад одна група розеток в одній кімнаті, тоді потрібно підвести лише один кабель до кімнати.

Визначення кількості пристроїв та типів пристроїв, що будуть підключатись до PLC

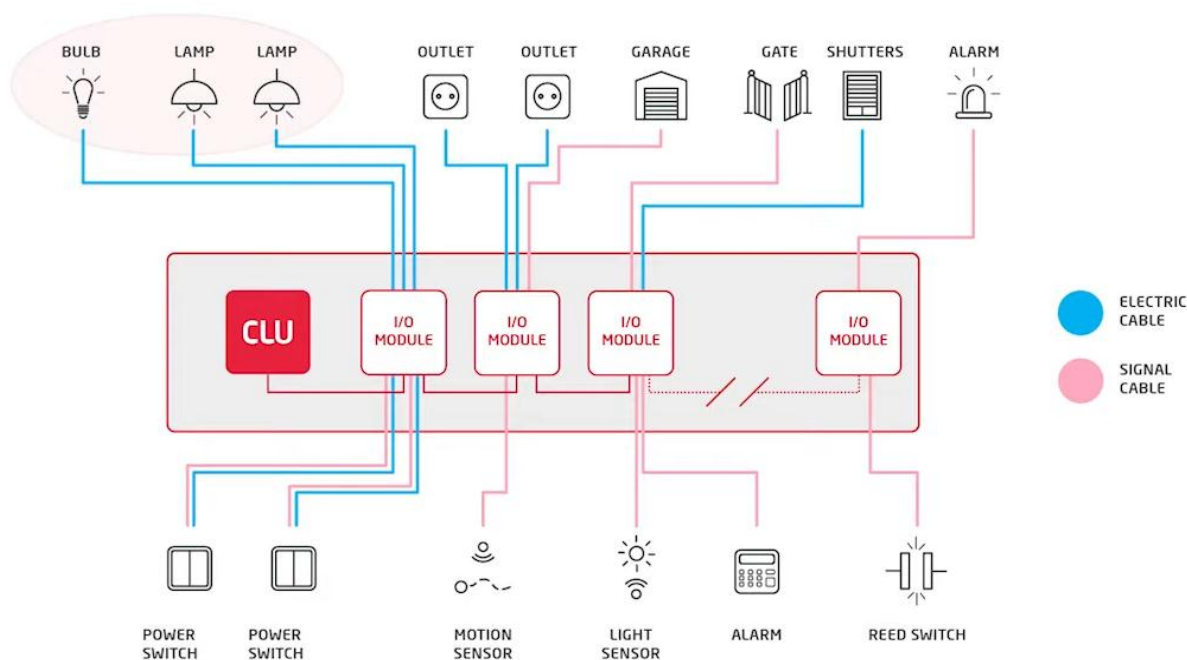


Рисунок 3.14 Приклад дротового розумного будинку

Керування гаражними воротами

Для керування гаражними воротами не потрібен силовий кабель, достатньо кабелю низької напруги. Поширеною практикою є використання витої пари (UTP). Усе що потрібно для відкриття, закриття чи зупинки гаражних воріт, подати сигнал низької напруги [29]. Аналогічно працює і відкриття та закриття в'їзних воріт.

Керування ролетними шторами

Керування шторами відрізняється, залежно від типу двигуна. На ринку існує два популярних типи двигунів. Двигуни змінного струму 220V та двигуни постійного струму 12 або 24V. Якщо на етапі будівництва невідомий тип двигуна, то найкраще організувати типовий електричний кабель у цьому випадку, чи це має бути дріт із поперечним перерізом 1,5 для освітлення.

Резервне живлення

Рекомендується розглядати використання резервного живлення для підтримки роботи системи у разі відмови основного живлення. Це допоможе уникнути перерв у функціонуванні та підтримати неперервну роботу системи у критичних ситуаціях.

Планування електромережі дротового розумного будинку

Зручність у вашому будинку слід планувати на етапі будівництва, зокрема, ретельно проектуючи всі інженерні мережі. Це визначає надійну роботу системи та забезпечує ваш спокій на тривалий термін.

Перш за все, це стосується планування електричної системи вашого приміщення. Важливо детально визначити, які прилади будуть використовуватися, де вони будуть розташовані та яка буде їхня потужність. З цієї інформації можна вибрати параметри живлення, зокрема, поперечний переріз провідника.

Переріз жили, КВ.ММ	Напруга, 220В	
	Струм, А	Потужність, кВт
1,5	19	4,1
2,5	27	5,9
4	38	8,3
6	46	10,1

Рисунок 3.15 визначення перерізу провідника

Існує кілька методик для правильного вибору поперечного перетину кабелів, включаючи прості формули. Наприклад, можна враховувати, що 1 мм² достатньо для струму 8-10А. Однак слід не тільки враховувати сумарний струм споживання, але й робити запас на майбутнє для забезпечення стабільної роботи та безпеки.

При проектуванні електромережі також важливо враховувати падіння напруги на довгих лініях через електричний опір кабелю. Для цього використовують формули або онлайн-калькулятори

Товщину перерізу дроту також можна визначити за допомогою формули. Для цього потрібно знати, яка різниця потенціалів (в вольтах) та максимальний струм (в амперах) може протікати через коло.

Матеріал: Вибір матеріалу є важливим, оскільки його опірна властивість (називається опором) є важливим фактором при визначенні поперечного перерізу (або товщини).

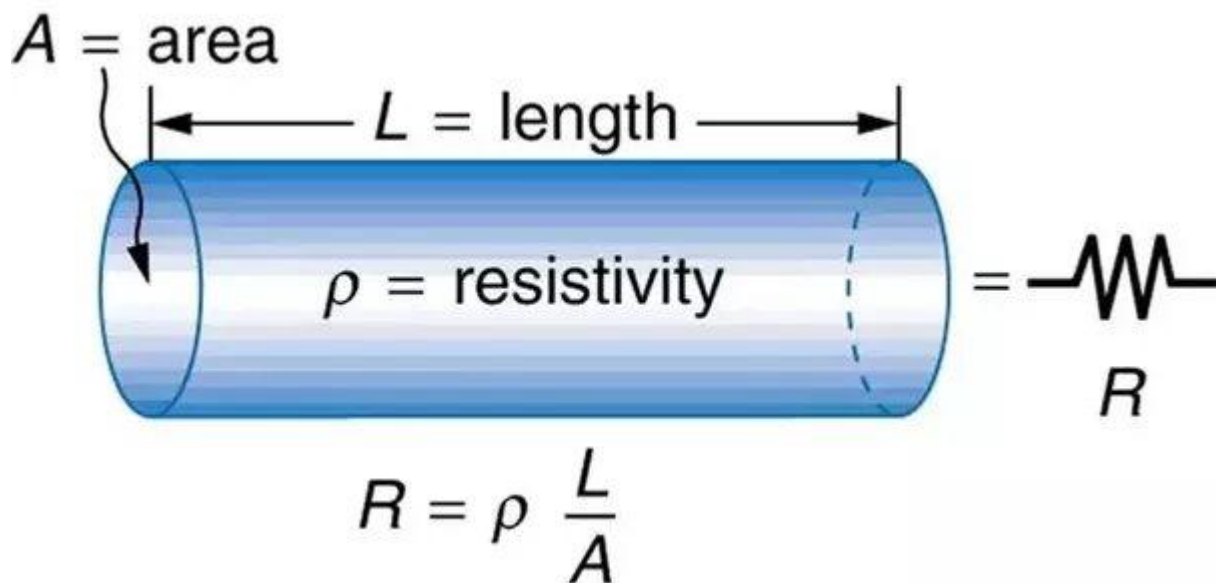


Рисунок 3.16 Формула розрахунку перерізу дроту

Неможливо мати дріт із нульовим опором. Тому розгляд опору є необхідним. Для прикладу припустімо, що дріт має опір 1 ом В залежності від довжини дроту (згідно з вашими вимогами), ви можете розрахувати поперечний переріз (або товщину дроту 'A') за такою формулою.

Поперечний переріз 'A' = [опірність (ρ) * Довжина 'L'] / Опір 'R'

Як вибрати правильне значення для опору 'R' при розрахунку товщини?

Оскільки електричне коло будуватиметься згідно з вимогами, необхідно бути впевненими щодо потужності навантаження тобто його споживання струму та напруги. Таким чином визначається потреба в струмі.

Отримана різниця потенціалів через опір дроту та течія через нього повинна бути непомітною в порівнянні з напругою джерела або акумулятора. Відповідно до цього вам потрібно враховувати опір.

ВИСНОВКИ

У ході даної дипломної роботи була визначена актуальність та детально проаналізовано нову лінійку промислових програмованих контролерів під назвою Arduino Opta.

Ці контролери відзначаються широким спектром можливостей програмування, які охоплюють лише традиційні мови, такі як IEC61131-3 Ladder та FBD PLC, але й сучасні інструменти, такі як C++ та Arduino IDE. Високий рівень надійності цих контролерів PLC відкриває шлях до застосування їх у створенні розумних будинків, відповідних індустріальним стандартам.

Дослідження альтернативних контролерів PLC на основі Arduino включало вивчення можливостей Arduino IDE, Arduino PLC та Arduino Cloud. Виявлено, що ці інструменти є потужними засобами новітнього програмування та дистанційного керування, відкриваючи перспективи для впровадження розумних рішень у побуті. Аналізувалися можливості інтеграції розумних пристроїв, зокрема управління мотором ролетних штор та системи енергомоніторингу, що дозволило отримати конкретні приклади застосування.

Отримані результати показують на потенціалі Arduino Opta як перспективне рішення для створення розумних будинків, що відповідають вимогам індустріальних стандартів. У висновках надані конкретні рекомендації щодо використання Arduino Opta в практиці, а також планування електромережі для ефективної роботи дротових розумних будинків.

Загалом, дослідження підкреслює деякі переваги використання Arduino Opta у сфері автоматизації та створення комфортних та енергоефективних розумних будинків.

ПЕРЕЛІК ПОСИЛАНЬ

1. <https://store.arduino.cc>
2. <https://store.arduino.cc/products/portentah7?queryID=49e390d016c6b29a1891304e41e931a6>
3. <https://www.arduino.cc/pro/hardware-product-family-portenta-family/>
4. <https://www.arduino.cc/pro/hardware-arduino-opta/>
5. <https://opta.findernet.com/en/> <https://docs.arduino.cc/hardware/opta>
6. <https://docs.arduino.cc/arduino-cloud/#arduino-iot-cloud-api>
7. <https://www.arduino.cc/pro/software-plc-ide/>
8. <https://docs.arduino.cc/arduino-cloud/guides/alexa>
9. <https://opta.findernet.com/en/tutorial/getting-started-connectivity>
10. https://www.academia.edu/86458793/Bluetooth_Based_Home_Automation_Using_Arduino
11. <https://opta.findernet.com/en/tutorial/plc-ide-setup-license>
12. <https://accautomation.ca/programming-the-arduino-opta-plc-ethernet-port/>
13. <https://opta.findernet.com/en/tutorial/using-plc-ide-with-arduino>
14. <https://docs.arduino.cc/tutorials/opta/energy-management-application-note>
15. <https://docs.arduino.cc/tutorials/opta/home-automation-application-note>
16. <https://cases.media/article/cikava-statistika-ta-fakti-pro-internet-rechei-iot-rozmir-rinku-vikoristannya-ta-prognozi>
17. <https://ts2.space/uk/роль-іот-в-автоматизації-розумного-будин/#gsc.tab=0>
18. <https://www.arduino.cc/en/software>
19. https://wirenboard.com/wiki/RS-485:_Wiring_and_Connection
20. <https://www.robo.house/uk/kursy-rozumnyy-budynok-i-mikrokontrolery-dlya-doroslykh/>
21. <https://itmaster.biz.ua/directory/standarts/spi.html>
22. <https://5watt.ua/uk/blog/statti/ssho-potribno-znati-pro-upravlinnya-osvitlennyam-dali>
23. https://duikt.edu.ua/uploads/p_2626_35882840.pdf
24. <https://docs.arduino.cc/hardware/opta>

25. <https://www.arduino.cc/pro/hardware-product-family-portenta-family/>
26. <https://doc.arduino.ua/ru/hardware/Uno>
27. <https://support.arduino.cc/hc/en-us/articles/360019833020-Download-and-install-Arduino-IDE>
28. <https://docs.arduino.cc/arduino-cloud/>
29. <https://grenton.pl/system/sterowanie/>
30. <https://docs.arduino.cc/tutorials/opta/energy-management-application-note>