

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр

Освітній рівень

Програмно-апаратний пристрій віддаленого керування системою відкриття
гаражних воріт на базі Arduino

Назва теми

КВРКІ.170294.17.02.24 ПЗ

Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія»

Назва

Виконав: студент IV курсу, група KI-17-2 О.В. Яцишин

Підпис

Ініціали, прізвище

Керівник

Підпис, дата

О.С. Андрощук

Ініціали, прізвище

Нормоконтролер

Підпис, дата

І.В. Муляр

Ініціали, прізвище

До захисту допускаю:

Зав. кафедри кібербезпеки та
комп'ютерних систем і мереж

Підпис

Ю.П.Кльоц

Ініціали, прізвище

« 25 » червня 2021 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ПРОГРАМУВАННЯ ТА КОМП'ЮТЕРНИХ І ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ

Кафедра КІБЕРБЕЗПЕКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ І МЕРЕЖ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЯ ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ю.П.Кльоц

“ 05 ” 02 2021 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Яцишину Олександрю Васильовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino

Керівник проекту (роботи) Андрощук О.С., д.т.н., проф.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 05.02.2021 р. № 11 додаток №7

2. Строк подання студентом проекту (роботи) на кафедру 29.06.2021 р.

3. Вихідні дані до проекту (роботи): розроблення програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____
Дослідження систем розумної автоматики для гаражних воріт як складової домашньої автоматизації та постановка задачі

Проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

Програмно-апаратна реалізація та тестування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

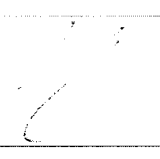
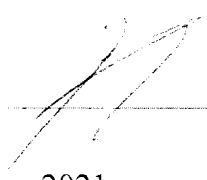

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) _____

Схема апаратних з'єднань Arduino Uno Rev3 та Ethernet Shield

Схема функціонування Ethernet Shield

Інтерфейсні вікна програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

6. Консультанти розділів дипломного проекту (роботи)


Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання приї
Нормоконтроль	Муляр І.В., доцент кафедри КБКСМ		
Антиплагіат	Муляр І.В., доцент кафедри КБКСМ		

7. Дата видачі завдання « 08 » 02 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Прим
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	11.01.2021	викон
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2021	викон
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2021	викон
4	Робота над розділом 2 – проектування підсистеми	01.04.2021	викон
5	Робота над розділом 3 – програмно-апаратна реалізація підсистеми	30.04.2021	викон
6	Оформлення пояснювальної записки згідно вимог	31.05.2021	викон
7	Попередній захист ВКР	07.06.2021	викон
8	Захист ВКР на засіданні ЕК	Червень 2021 року	

Студент


Підпис

О.В. Яцишин
Ініціали, прізвище

Керівник проекту (роботи)


Підпис

О.С. Андрощук
Ініціали, прізвище

№ р я д к а	ф о р м а т	Позначення	Найменування	К і л л и с т і в	№ ек з	П р и м і т к а
			<u>Текстові документи</u>			
1		КВРКІ.170294.17.02.24 ПЗ	Пояснювальна записка	69		
			<u>Графічні матеріали</u>			
2		КВРКІ.170294.17.02.24 Е8	Схема апаратних з'єднань Arduino Uno Rev3 та Ethernet Shield	1		
3		КВРКІ.170294.17.02.24 Е8	Схема функціонування Ethernet Shield	1		
4		КВРКІ.170294.17.02.24 Е8	Інтерфейсні вікна програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino	1		
КВРКІ.170294.17.02.24 ВП						
Зм	Арк	№ докум	Підпис	Дата		
Розробив		Яцишин О.В.			Літера	Аркуш
Перевір.		Андрощук О.С.			н	1
Н. контр.		Муляр І.В.			ХНУ, КІ-17-2	
Затв.		Кльод Ю.П.				

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino».

Автор роботи: Яцишин Олександр Васильович.

Керівник роботи: Андрощук Олександр Степанович.

Пояснювальна записка: 69 с., 37 рис., 5 табл., 4 дод., 40 джерел.

Графічна частина: 8 презентаційних слайдів.

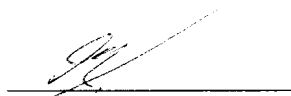
СИСТЕМА ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ, СИСТЕМА ВІДКРИТТЯ ГАРАЖНИХ ВОРІТ, ARDUINO, СЕРВОПРИВІД, ІНТЕРНЕТ РЕЧЕЙ, ДАВАЧ.

Метою роботи є проектування та реалізація програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Об'єктом дослідження є програмно-технічний (апаратний) засіб – програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Предметом дослідження є програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Практична цінність роботи полягає в спроектованому та реалізованому програмно-апаратному пристрої віддаленого керування системою відкриття гаражних воріт на базі Arduino, в якому враховано недоліки комерційних систем розумної автоматки для гаражних воріт та який може бути застосований як складова частина в системах домашньої автоматизації.



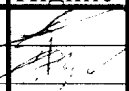
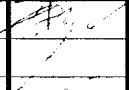

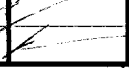
Підпис студента

25.06.2021

Дата

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ.....	4
ВСТУП.....	5
1 ДОСЛІДЖЕННЯ СИСТЕМ РОЗУМНОЇ АВТОМАТИКИ ДЛЯ ГАРАЖНИХ ВОРІТ ЯК СКЛАДОВОЇ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ ТА ПОСТАНОВКА ЗАДАЧІ.....	7
1.1 Функції систем домашньої автоматизації	7
1.2 Реалізація системи домашньої автоматизації на базі Arduino.....	9
1.3 Апаратна платформа Arduino.....	11
1.4 Сервоприводи як складові систем розумної автоматики для гаражних воріт	18
1.5 Огляд комерційних систем розумної автоматики для гаражних воріт	22
1.6 Висновки. Постановка задачі.....	25
2 ПРОЕКТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ПРИСТРОЮ ВІДДАЛЕНОГО КЕРУВАННЯ СИСТЕМОЮ ВІДКРИТТЯ ГАРАЖНИХ ВОРІТ НА БАЗІ ARDUINO	27
2.1 Обґрунтування вибору плати Arduino Uno Rev 3.....	27
2.2 Обґрунтування вибору плати Ethernet Shield.....	29
2.3 Обґрунтування вибору програмного забезпечення для програмування плати Arduino Uno Rev 3	31
2.4 Підключення плат керування Ethernet Shield та Arduino Uno та тестування їх спільної роботи.....	35
2.5 Висновки	46
3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ПРИСТРОЮ ВІДДАЛЕНОГО КЕРУВАННЯ СИСТЕМОЮ ВІДКРИТТЯ ГАРАЖНИХ ВОРІТ НА БАЗІ ARDUINO.....	47
3.1 Модифікація комутатора панелі керування сервоприводом.....	47

					КВРКІ.170294.17.02.24ПЗ			
№	Арк	№докум.	Підпис	Дата	Програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino Пояснювальна записка	Літера	Аркуш	Аркушів
Виконав		Яцишин О.В.					2	69
Перевір.		Андрощук О.С.						
Контр.		Муляр І.В.						
Звер.		Кльоц Ю.П.						ХНУ, КІ-17-2

3.2 Створення прототипу програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino	49
3.3 Використання середовища Visual Studio 2019 IDE для написання прошивки плати Arduino	53
3.4 Тестування коректності завантаження прошивки через середовище VS2019.....	57
3.5 Написання модифікованої прошивки для програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino	59
3.6 Тестування коректності роботи модифікованої прошивки програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino	63
3.7 Висновки	66
ВИСНОВКИ	67
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	70
Додаток А Копія креслення «Схема апаратних з'єднань Arduino Uno Rev3 та Ethernet Shield»	73
Додаток Б Копія креслення «Схема функціонування Ethernet Shield»	74
Додаток В Копія креслення «Інтерфейсні вікна програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino»	75
Додаток Г Лістинг коду для прошивки плати Arduino	76

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

АЦП – аналого-цифровий перетворювач

MAC – Media Access Control

ОС – Операційна система

ПЗ – Програмне забезпечення

ШІМ – Широтно-імпульсна модуляція

DUT – Device under test

EEPROM – Electrically Erasable Programmable Read-Only Memory

FTDI – Future Technology Devices International

GND – Ground

GUI – Graphical User Interface

HTML – HyperText Markup Language

ICSP – In-Circuit Serial Programming

IDE – Integrated Development Environment

IOT – Internet of Things

IP – Internet Protocol

LAN – Local Area Network

NPN – Negative-Positive-Negative

PDM – Pulse Duration Modulation

UART – Universal asynchronous receiver/transmitter

USB – Universal Serial Bus

WPA – Wi-Fi Protected Access

					КВРКІ.170294.17.02.24 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		4

ВСТУП

На сьогоднішній день в повсякденне життя людини активно впроваджуються інноваційні технології у вигляді систем домашньої автоматизації [1, 2], що надають можливість істотно підвищити комфорт, безпеку та заощадити час, витрачений на виконання різних побутових задач. Система домашньої автоматизації (англ. *home automation*) надає можливість управляти та контролювати роботою будь-яких механізмів у домі – шторами, жалюзями, воротами, вхідними дверима, гаражними воротами та шлагбаумами, кранами води і газу тощо [3].

Підвищення рівня безпеки [4, 5] та комфорту мешканців і доступності використання будинкового обладнання є важливим завданням також з точки зору вирішення основних соціальних проблем. Крім того, що системи домашньої автоматизації підвищують загальний рівень комфорту, можливість дистанційного керування механізмами є важливою функцією для літніх людей та людей з обмеженими можливостями.

Основними недоліками комерційних систем домашньої автоматизації є складність монтажу та завищена вартість таких систем, а також неможливість додавання власних функцій управління і безпеки [6].

Тому актуальною є задача проектування власних пристроїв для домашньої автоматизації, які б не поступалися за функційними характеристиками комерційним системам, але при тому дозволяли б усунути перераховані вище недоліки.

Метою роботи є проектування та реалізація програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Об'єктом дослідження є програмно-технічний (апаратний) засіб - програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Предметом дослідження є програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						5
Зм.	Арк.	№докум.	Підпис	Дата		

Практична цінність роботи полягає в спроектованому та реалізованому програмно-апаратному пристрої віддаленого керування системою відкриття гаражних воріт на базі Arduino, в якому враховано недоліки комерційних систем розумної автоматики для гаражних воріт та який може бути застосований як складова частина в системах домашньої автоматизації.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		6

1 ДОСЛІДЖЕННЯ СИСТЕМ РОЗУМНОЇ АВТОМАТИКИ ДЛЯ ГАРАЖНИХ ВОРІТ ЯК СКЛАДОВОЇ ДОМАШНЬОЇ АВТОМАТИЗАЦІЇ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Функції систем домашньої автоматизації

На сьогоднішній день у побуті набувають все більшої популярності системи домашньої автоматизації як багатофункціональні системи управління механізмами в будинку (на виробництві, квартирі або офісі). Інша назва таких систем – «розумний будинок» (англ. *smart house*) [1-6].

Автоматизоване управління має переваги перед звичайними механізмами. Змістовна перевага – це можливість дистанційного контролю та управління системою домашньої автоматизації на відстані. Використання спеціалізованого пульта управління дає змогу легко контролювати роботу таких механізмів – відкривати або закривати жалюзі, гаражні ворота, вікна, ворота, двері тощо, дистанційно перекривати крани води і газу, навіть перебуваючи фізично поза межами будинку. Ця перевага стає ще більш очевидною для літніх людей, людей з обмеженими можливостями, або, наприклад, коли вікна в приміщенні знаходяться занадто високо чи доступ до них пергороджений меблями.

Системи домашньої автоматизації вже тривалий час є невід'ємною частиною організації побуту для багатьох будинків, квартир і офісних приміщень.

Система домашньої автоматизації надає можливості реалізувати безліч різноманітних функцій [1-3]:

- 1) кліматконтроль як реакція на зміну погодних умов і температури в приміщенні та увімкнення/вимкнення відповідних пристроїв – опалювальних, вентиляційних, кондиціонерів тощо;
- 2) відслідковування стану вікон і дверей – відкриті вони чи закриті;
- 3) функція сигналізації – звукове сповіщення при спрацюванні датчика руху;
- 4) автоматичне управління побутовою технікою;

					КВРКІ.170294.17.02.24 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		7

5) контроль споживання електроенергії шляхом автоматичного увімкнення/вимкнення освітлення [7];

6) забезпечення пожежної безпеки – звукові сповіщення при наявності спалаху або диму в приміщенні та виклик пожежної служби;

7) керування меблями, інтер'єрними предметами;

8) керування порт'єрами, шторами, ролетами, жалюзіями;

9) керування екранами, відеопроєкторами, телевізорами;

10) керування вікнами, дверима, воротами, шлагбаумами.

Система домашньої автоматизації працює спільно з системою механізованих елементів, давачів та фурнітури різного призначення. Особливою зручністю є те, що керувати системами домашньої автоматизації можна не лише за допомогою спеціалізованих пультів управління та сенсорних панелей, а й з допомогою комп'ютера, ноутбука чи смартфона через мережу Інтернет.

При цьому керування механізованими елементами виконується автоматично з використанням давачів, які вбудовані в карниз. Необхідні команди подаються через панель управління, після чого система домашньої автоматизації керує роботою модуля згідно визначених умов.

Системи автоматичного управління шторами, жалюзі та ролетами можуть бути використані як в приватних будинках, квартирах, котеджах, так і в офісах, на промислових та виробничих об'єктах. Автоматизація подібних задач дозволяє створити сприятливі та комфортні умови для життя та роботи, забезпечити і заощадити час на користь продуктивності праці співробітників.

Зокрема, система розумної автоматизації для гаражних воріт складається з:

1) блоку управління;

2) давачів;

3) фотоелементів;

4) пульта, завдяки якому здійснюється дистанційне керування воротами (при натисканні кнопки на пульті автоматика воріт приймає сигнал та виконує відповідну команду).

Спеціалізовані механізми та фурнітура для воріт слугують також для запобігання можливості контакту стулок дверей з транспортом під час в'їзду та

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		8

виїзду автотранспорту. Давачі зазвичай встановлюються на відстані 30 – 90 см від поверхні землі та спрацьовують під час закривання воріт. Якщо виявлено будь-яку перешкоду – автомобіль, людину чи тварину, то автоматично припиняється процес закриття і відкриття воріт або шлагбаума і продовжується лише тоді, коли буде усунена ця перешкода.

1.2 Реалізація системи домашньої автоматизації на базі Arduino

Використання платформи Arduino для проектування систем домашньої автоматизації надає важливу перевагу: можливість збору системи з окремих складових на зразок конструктора [8-10]. Така система передбачає наявність центрального елемента на платформі Arduino, представленого центральним контролером, на який передається наявна інформація від різноманітних підсистем, встановлених в будинку. По мірі необхідності в систему домашньої автоматизації можна додавати нові компоненти, які виконуватимуть різні функції автоматизації в будинку: управляти освітленням, при виникненні непередбачених ситуацій сповіщати про них власника будинку, контролювати кліматичні умови в будинку, стежити за різними інженерними механізмами, відкривати та закривати вікна, жалюзі, гаражні ворота або здійснювати автоматичний полив газонів. Модульність такої системи дозволяє використовувати на початку лише необхідні компоненти і в міру необхідності нарощувати потрібний функціонал.

Існує велика кількість комплектацій на базі Arduino [11-13], на основі яких може бути реалізована така система домашньої автоматизації. Виробники випускають контролери з вбудованими системами Bluetooth та Wi-Fi, що надає можливість управляти системою домашньої автоматизації через мобільний Інтернет. Також управління автоматизованою системою може здійснюватися з використанням Ethernet, тобто шляхом провідного підключення за допомогою оптоволоконних кабелів через домашню локальну мережу. Підключення вимикачів, що керуються автоматизованою системою на базі Arduino, можна здійснювати за допомогою електричної проводки та за бездротовими технологіями.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						9
Зм.	Арк.	№докум.	Підпис	Дата		

Система домашньої автоматизації на базі Arduino може складатися з наступних компонентів [14]:

- 1) центральний контролер, зазвичай представлений головним вузлом;
- 2) пристрої зв'язку та розширення (наприклад, роутери, комутатори, модулі GPS і GPRS);
- 3) прилади, що відповідають за комутацію електроланцюгів (наприклад, реле, диммери, а також блоки живлення);
- 4) прилади виконання (наприклад, клапани газу, води);
- 5) складові для системного управління (сенсорні панелі, планшети, персональні комп'ютери або пульти);
- б) вимірювальні складові (давачі і сенсори).

При виборі обладнання для систем домашньої автоматизації на базі Arduino необхідно враховувати, який метод передачі даних буде використовувати система.

До переваг реалізації систем домашньої автоматизації на базі Arduino можна віднести наступні [15, 16]:

- 1) широкі можливості в плані налаштування роботи системи домашньої автоматизації, оскільки самостійно можна написати програму, яка виконуватиме алгоритми різного рівня складності;
- 2) можливість автономного функціонування при наявності власного контролера;
- 3) простота завантаження програм, оскільки немає необхідності у використанні спеціальних програматорів;
- 4) низька вартість комплектуючих;
- 5) наявність відкритого вихідного коду, що надає можливість безпосередньо користувачеві управляти механізмами системи домашньої автоматизації;
- б) доступність та прозорість, адже користувач сам може обирати, які складові, механізми та давачі йому необхідні;
- 7) універсальність і можливість реалізації будь-яких ідей, оскільки користувач може створити систему автоматизації за власними вимогами;
- 8) можливість самостійної прошивки завантажувача;

					КВРКІ.170294.17.02.24 ПЗ	Арк
						10
Зм.	Арк.	№докум.	Підпис	Дата		

9) наявність спеціального штирьового роз'єму у плат Arduino, що надає можливість виконувати програмування всередині системи домашньої автоматизації.

1.3 Апаратна платформа Arduino

Arduino Uno є найбільш поширеною стандартною платою Arduino [11]. Плата Arduino Uno Rev3 [17] базується на мікроконтролері ATmega328P. Загальний вигляд Arduino Uno Rev3 подано на рисунку 1.1. Розглянемо основні характеристики Arduino Uno Rev3. Платформа містить в собі 14 цифрових пінів входу та виходу. 6 з цих пінів використовуються як виходи ШІМ. Також платформа містить 6 аналогових входів, роз'єми USB, силовий, ICSP, кварцовий генератор з частотою 16 МГц та кнопку перезавантаження.

Платформа підключається до комп'ютера через кабель USB, або ж живлення подається за допомогою батареї чи адаптера AC/DC.



Рисунок 1.1 – Загальний вигляд плати Arduino Uno Rev3 з двох сторін

Відмінність від попередніх плат Arduino Uno полягає в тому, що в якості перетворювача інтерфейсів USB–UART використовується мікроконтролер ATmega16U2 (ATmega8U2 до версії R2) замість мікросхеми FTDI. В варіантах китайського виробництва використовується перетворювач інтерфейсів USB–UART CH340G.

В версії Arduino Uno R2 для полегшення процесу оновлення прошивки доданий резистор, що підключає вивід HWB мікроконтролера 8U2 до землі.

Відмінності, характерні для версії R3, є наступними:

1) розпіновка 1.0: додані виводи SCL та SDA (біля виводу AREF), а також два виводи біля виводу RESET: IOREF, що дозволяє платам розширення підлаштовуватися під робочу напругу Arduino, що передбачений для сумісності плат розширення як з Arduino з 5 В на базі мікроконтролерів AVR, так і з платами Arduino Due з 3.3 В; другий вивід зарезервованій для майбутніх цілей;

2) удосконалена стійкість щодо завад ланцюга скидання;

3) мікроконтролер версії ATmega8U2 змінений на ATmega16U2.

Ці та інші основні характеристики плати Arduino Uno Rev3 подано в таблиці 1.1.

Таблиця 1.1 – Основні характеристики плати Arduino Uno Rev3

Характеристика	Параметри
Мікроконтролер	ATmega328P
Робоча напруга	5В
Напруга живлення (рекомендована)	7–12В
Напруга живлення (гранична)	6–20В
Цифрові входи / виходи	14 (з них 6 можуть застосовуватись в ролі ШІМ–виходів)
Аналогові входи	6
ШІМ (PWM) Піни	6
Постійний струм через вхід / вихід	40 мА
Максимальний вихідний струм виводу 3.3V	50 мА
Flash–пам'ять	32 КБ (ATmega328P) з них 0.5 КБ застосовуються завантажником
SRAM	2 КБ (ATmega328P)
EEPROM	1 КБ (ATmega328P)

Продовження таблиці 1.1 – Основні характеристики плати Arduino Uno Rev3

Характеристика	Параметри
Тактова частота	16 МГц
Вбудований світлодіод	13
Довжина	68.6 мм
Ширина	53.4 мм
Вага	25 г

Розглянемо особливості плати Arduino Uno CH340G Rev3, загальний вигляд якої подано на рисунку 1.2. Плата Arduino Uno CH340G Rev3 характеризується всіма властивостями плати Arduino Uno Rev3. Відмінності від Arduino Uno Rev3 пов'язані з зменшенням вартості Arduino Uno CH340G Rev3 за рахунок застосування мікроконтролера ATmega328 в корпусі SMD. Також міст USB виконаний на основі бюджетної мікросхеми CH340G. Елементи зовнішніх підключень та імена портів, на цій платі відповідають принциповій схемі Arduino Uno Rev3. Проте нумерація корпусів DIP та виводів SMD мікроконтролера ATmega328 відрізняється, як видно з рисунку 1.3.



Рисунок 1.2 – Загальний вигляд плати Arduino Uno CH340G Rev3



Рисунок 1.3 – Відмінності між Arduino Uno Rev3 та Arduino Uno CH340G Rev3

Для мікросхеми моста USB CH340G необхідно встановити драйвери, які можна скачати на офіційному сайті [17].

Розглянемо більш детально призначення основних елементів плати Arduino Uno Rev3, відображених на рисунку 1.4:

- 1) USB Plug – роз'єм для підключення USB-пристроїв;
- 2) Analog Reference Pin – роз'єм, що застосовується для визначення опорної напруги АЦП;
- 3) Digital Ground – земля;
- 4) Digital I / O Pins (2–13) – цифрові виводи;
- 5) Serial OUT (TX) – пін для передачі даних по UART;
- 6) Serial IN (RX) – пін для прийому даних по UART;
- 7) Reset Button – кнопка для перезавантаження мікроконтролера;
- 8) In-Circuit Serial Programmer (ISCP) – контакти для перепрограмування плати;
- 9) ATmega328P Microcontroller – чіп;
- 10) Analog In Pins (0-5) – аналогові входи;
- 11) Voltage In – вхід, що використовується для подачі живлення від зовнішнього джерела живлення;
- 12) Ground Pins – земля;
- 13) 5 Volt Power Pin – живлення 5 В;
- 14) 3 Volt Power Pin – живлення 3.3 В;

15) Reset Pin – вхід для перезавантаження;

16) External Power Supply – роз'єм підключення зовнішнього джерела живлення.

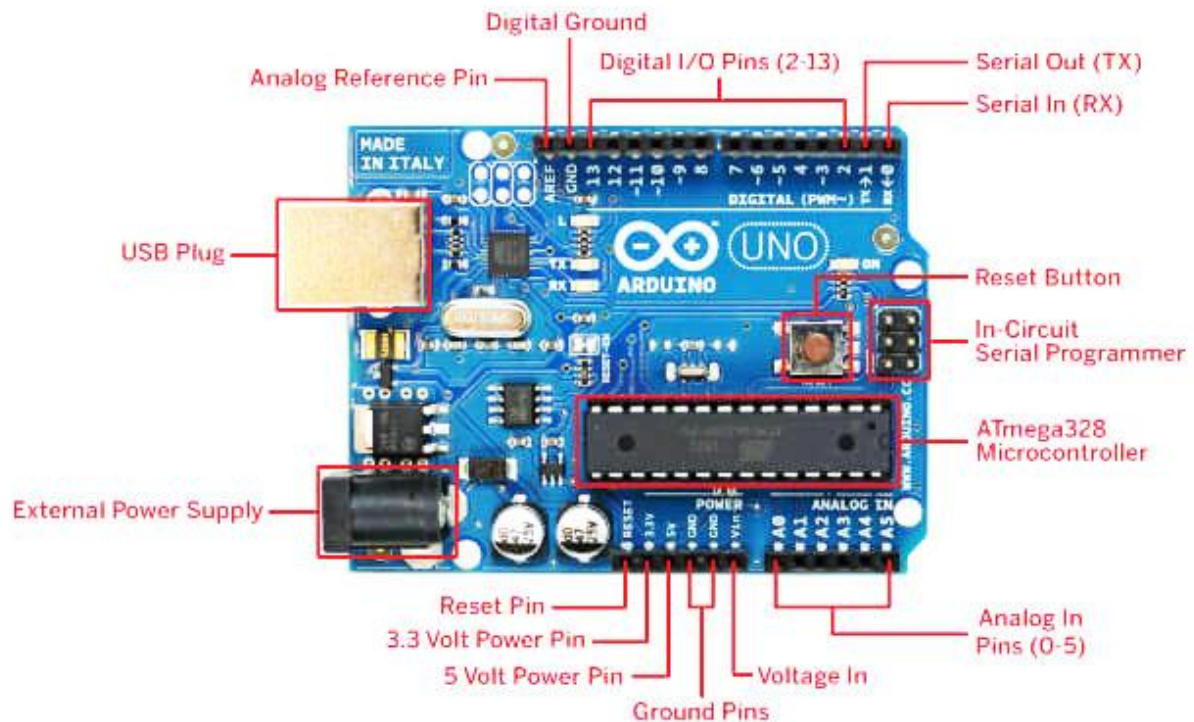


Рисунок 1.4 – Основні елементи плати Arduino Uno Rev3 [17]

Піни Arduino Uno Rev3 можуть бути використані для підключення зовнішніх пристроїв та працюють в режимі входу і виходу. Кожен вивід має навантажувальний резистор, який за замовчуванням відключений, з опором 20-50 кОм, який здатен пропускати струм до 40 мА.

Деякі виводи Arduino Uno Rev3 мають особливе призначення:

- 1) піни 0 і 1 – виконують функції контактів UART (відповідно RX і TX);
- 2) піни 10 - 13 – виконують функції контактів SPI (відповідно SS, MOSI, MISO і SCK);
- 3) піни A4 і A5 – виконують функції контактів I2C (відповідно SDA і SCL).

Піни 0 – 13 є цифровими, тобто можна зчитувати і подавати на них лише сигнали HIGH і LOW. За допомогою ШІМ для управління потужністю підключених пристроїв можна використовувати цифрові порти. Адресація та призначення цифрових пінів 0-13 подані в таблиці 1.2.

Таблиця 1.2 – Адресація та призначення цифрових пінів 0-13

Пін	Адресація	Спеціальне призначення	ШІМ
0	0	RX	
1	1	TX	
2	2	Вхід для переривань	
3	3	Вхід для переривань	ШІМ
4	4		
5	5		ШІМ
6	6		ШІМ
7	7		
8	8		
9	9		ШІМ
10	10	SPI (SS)	ШІМ
11	11	SPI (MOSI)	ШІМ
12	12	SPI (MISO)	
13	13	SPI (SCK), містить вбудований світлодіод	

Аналогові піни Arduino Uno Rev3 слугують для підключення аналогових пристроїв та в якості входів для вбудованого десятирозрядного аналого-цифрового перетворювача (АЦП). Адресація та призначення аналогових пінів Arduino Uno Rev3 наведені в таблиці 1.3.

Таблиця 1.3 – Адресація та призначення аналогових пінів Arduino Uno Rev3

Пін	Адресація	Спеціальне призначення
A0	A0 або 14	
A1	A1 або 15	
A2	A2 або 16	
A3	A3 або 17	
A4	A4 або 18	I2C (SCA)
A5	A5 або 19	I2C (SCL)

Також плата Arduino Uno Rev3 оснащена додатковими пінами. Пін AREF видає опорну напругу для вбудованого аналогово-цифрового перетворювача та може бути керований за допомогою функції `analogReference()`. Пін RESET передає низький рівень сигналу на вивід та перезавантажує мікроконтролер. Зазвичай цей пін застосовується для підключення кнопки перезавантаження, що закриває доступ до такої кнопки безпосередньо на платі Arduino.

Плати Arduino Uno Rev3 можуть житись через підключення USB, а також від зовнішнього джерела живлення. При цьому джерело живлення обирається автоматично.

Можливі наступні способи живлення плати Arduino Uno R3:

- 1) від зовнішнього адаптера, при цьому рекомендована напруга складає 7 - 12В.
- 2) від порту USB комп'ютера;
- 3) подача 5В безпосередньо на пін 5V, проте в цьому випадку не задіяний вхідний стабілізатор, тому навіть незначне перевищення напруги може спричинити вихід плати Arduino Uno Rev3 з ладу.

Якщо напруга перевищує 12В при живленні від зовнішнього адаптера, то регулятор напруги може перегрітися, що також призведе до пошкодження плати. Якщо напруга живлення нижча 7В, до прикладу 5В, то вивід 5V може видавати менше 5В, що може спричинити нестабільну роботу плати Arduino Uno Rev3.

Розглянемо призначення виводів живлення:

- 1) 5В – на цей пін Arduino Uno Rev3 подає 5В, його можна використовувати для живлення зовнішніх пристроїв;
- 2) 3.3V – на цей пін подається напруга 3.3 В від внутрішнього стабілізатора;
- 3) GND – вивід землі;
- 4) VIN – пін, що використовується для подачі зовнішньої напруги;
- 5) IREF – пін, що використовується для передачі зовнішнім пристроям даних про робочу напругу плати Arduino Uno Rev3.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		17

1.4 Сервоприводи як складові систем розумної автоматики для гаражних воріт

Важливою складовою системи розумної автоматики для гаражних воріт є сервопривід. Сервопривід – це мотор з керуванням через негативний зворотний зв'язок, що дозволяє точно керувати параметрами руху. Сервомотором є механічний привод будь-якого типу, що містить в своєму складі давач положення і плату управління [18, 19].

Тобто сервопривід – це механізм з електромотором, який може повертатися на заданий кут, а також утримувати поточний стан.

За перетворення електрики на механічний оберт в сервоприводі відповідає електромотор. В асинхронних сервоприводах використовується колекторний мотор, а в синхронних сервоприводах – безколекторний.

Швидкість обертання мотора зазвичай занадто велика для практичного використання, а обертовий момент занадто слабкий. Для вирішення зазначених проблем використовується редуктор, який є механізмом з шестернею, що передає і перетворює обертовий момент.

Коли вмикається і вимикається електромотор, то обертається вихідний вал, який є кінцевою шестернею редуктора, до якої можна приєднати щось, чим потрібно керувати (в нашому випадку – гаражні ворота).

З метою контролю положення валу використовується встановлений на сервоприводі давач зворотного зв'язку. Це може бути потенціометр або енкодер. Позиціонер перетворює кут оберту валу назад в електричний сигнал.

За обробку даних в сервоприводі відповідає плата управління, яка порівнює одержане з мікроконтролера значення з показником давача зворотного зв'язку, і згідно із результатом порівняння включає або вимикає мотор відповідно.

Вал є частиною редуктора, що виведена за межі корпусу мотора і приводиться в рух безпосередньо при подачі на сервопривід керуючих сигналів. У комплектації сервомоторів також включені гойдалки різних формфакторів, які з'єднуються з валом сервоприводу [19].

					КВРКІ.170294.17.02.24 ПЗ	Арк
						18
Зм.	Арк.	№докум.	Підпис	Дата		

При цьому потрібно враховувати, що додавати до валу навантаження, які більші за обертовий момент сервоприводу, не рекомендується. Це може призвести до руйнування редуктора сервоприводу.

Для роботи сервоприводу його необхідно підключити до джерела живлення, а також до керуючої плати. Сервопривід оснащений шлейфом з трьох проводів:

1) червоний – живлення сервомотора, що підключається до плюсового контакту джерела живлення;

2) чорний – земля, що підключається до мінусового контакту джерела живлення і землі мікроконтролера;

3) жовтий – керуючий сигнал, що підключається до цифрового піну мікроконтролера.

Якщо сервопривід живиться напругою від 5 В і споживає струм, менший за 500 мА, то зовнішнє джерело живлення можна не використовувати і безпосередньо підключити провід живлення сервомотора до джерела живлення мікроконтролера.

Алгоритм роботи сервоприводу наступний:

1) Сервопривід на вхід отримує керуючі імпульси, які містять наступні значення: для простих сервоприводів – значення кута оберту, для сервоприводів постійного обертання – значення напрямку та швидкості обертання.

2) Плата управління порівнює одержані значення з показниками давача зворотного зв'язку.

3) На основі одержаного результату порівняння привід виконує визначену дію: поворот, прискорення або уповільнення таким чином, щоб значення, одержане з внутрішнього давача, стало найбільш близьким до значення зовнішнього керуючого параметра.

Інтерфейс управління сервоприводу реалізований наступним чином: для того, щоб вказати сервоприводу потрібний стан, необхідно надіслати керуючий сигнал по сигнальному проводу – імпульси постійної частоти і змінної ширини.

Положення, яке повинен зайняти сервопривід, залежить безпосередньо від довжини переданих імпульсів. Коли в керуючу схему сервоприводу надходить сигнал від мікроконтролера, то генератор імпульсів виробляє імпульс, тривалість

					КВРКІ.170294.17.02.24 ПЗ	Арк
						19
Зм.	Арк.	№докум.	Підпис	Дата		

якого визначається через давач зворотного зв'язку. Після цього схема порівнює тривалість цих двох імпульсів. Якщо тривалість імпульсів різна, то включається електромотор з певним напрямком обертання, що визначається тим, який з імпульсів був коротшим. Якщо ж довжини імпульсів є рівними, то електромотор зупиняється.

Для управління сервоприводами використовуються імпульси з частотою 50 Гц, тобто їх період дорівнює 20 мс, при цьому 1540 мкс означає, що сервопривід повинен зайняти середнє положення, 544 мкс – для 0° , а 2400 мкс визначає оберт на 180° . Принцип управління сервоприводом за допомогою імпульсів різної ширини відображено на рис. 1.5 [19].

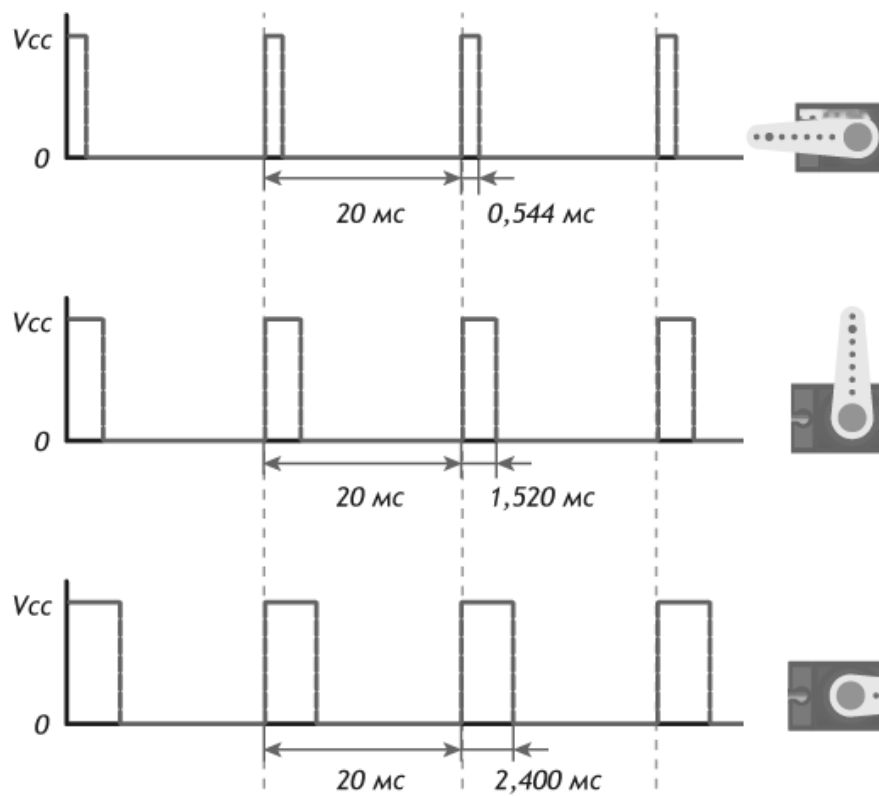


Рисунок 1.5 – Принцип управління сервоприводом за допомогою імпульсів різної ширини

Проте деякі сервоприводи використовують нестандартну ширину імпульсу – 760 мкс, що залежить від заводських налаштувань. Середнє положення в такому випадку відповідає 760 мкс, тоді як в звичайних сервоприводах середньому положенню відповідає 1 520 мкс.

При виробництві кожної моделі сервоприводу припускається можливість наявності певної похибки, що призводить до того, що робочий діапазон довжин імпульсів може відрізнятись від еталонного. Тому кожен сервопривід обов'язково повинен бути відкалібрований для більш точної роботи. Калібрування сервоприводу полягає в визначенні експериментальним шляхом коректного діапазону імпульсів, характерного саме для цього конкретного сервоприводу.

Спосіб управління сервоприводами одержав назву PDM (Pulse Duration Modulation), тому що при управлінні важлива саме довжина імпульсів, а не частота.

Основні характеристики сервоприводів наступні:

- 1) обертаючий момент;
- 2) швидкість повороту;
- 3) форм-фактор (мікро, стандартний, великий);
- 4) внутрішній інтерфейс (аналогові та цифрові);
- 5) матеріал, з якого виготовлено шестерні (пластик, карбон, метал);
- 6) тип мотору (колекторний або безколекторний);
- 7) кут обертання (обмежений сервопривід – одержав назву «сервопривід 180°»; або необмежений сервопривід – одержав назву сервопривід постійного обертання чи «сервопривід 360°»).

При проектуванні систем розумної автоматики для гаражних воріт більшість сервоприводів можуть бути підключені до плати Arduino безпосередньо через спеціальний шлейф, що складається з трьох проводів [20]:

- 1) червоний – живлення, що підключається або до контакту 5V або до джерела живлення безпосередньо;
- 2) коричневий або чорний – земля;
- 3) жовтий або білий – сигнал, що підключається до цифрового виходу плати Arduino.

Для підключення сервоприводу до плати Arduino можна скористатися платою-розширником портів, або ж безпосередньо підключити сервопривід до контактів Arduino. Також можна генерувати імпульси самостійно з використанням стандартної бібліотеки Servo [21].

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		21

Звичайний сервопривід споживає під час роботи більше 100 мА. При цьому плата Arduino здатна видавати струм до 500 мА. Тому, якщо виникає необхідність використовувати потужний сервопривід, необхідно підключити його в контур з додатковим живленням.

Для переважної більшості плат Arduino бібліотека Servo підтримує управління не більше ніж 12 сервоприводами. Проте Arduino Mega дозволяє керувати 48 сервоприводами.

При використанні бібліотеки Servo потрібно враховувати важливу особливість [21]: при використанні відмінної від Arduino Mega плати функцію analogWrite() неможливо використовувати на 9 та 10 контактах незалежно від того, підключені чи не підключені сервоприводи до цих контактів. На Arduino Mega існує можливість підключення до 12 сервоприводів без порушення функціонування ШІМ/PWM. Проте при використанні більшої кількості сервоприводів на Arduino Mega неможливо використовувати функцію analogWrite() на 11 та 12 контактах.

1.5 Огляд комерційних систем розумної автоматички для гаражних воріт

Існує багато готових комерційних рішень розумної автоматички для гаражних воріт.

Одним з таких рішень є система для керування воротами італійського виробництва Comunello Rampart [22].

Основними характеристиками системи є наступні:

- 1) наявність світлодіодного блоку;
- 2) наявність блоку управління з однією кнопкою;
- 3) налаштування різноманітних параметрів руху гаражних воріт;
- 4) можливість вибору різних режимів управління гаражними воротами;
- 5) наявність механізму розблокування;
- 6) наявність функції автозакриття;
- 7) можливість регулювання робочого зусилля приводу;
- 8) блок управління функціями виявлення перешкод.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		22

Зовнішній вигляд системи для керування воротами Comunello Rampart RT1000LKIT наведено на рисунку 1.6.



Рисунок 1.6 – Зовнішній вигляд системи для керування воротами Comunello Rampart RT1000LKIT

Іншим популярним рішенням є система домашньої автоматизації HomeKit від Apple [23], яка в числі стандартних функцій домашньої автоматизації містить також функціонал керування гаражними воротами. Інтерфейс системи HomeKit наведено на рисунку 1.7.

Комерційне рішення німецького виробництва Hörmann [24] для керування гаражними воротами надає можливість дистанційного керування з мобільного телефону через Bluetooth чи за допомогою пульта керування. Функція дистанційного керування реалізована на базі радіоз'єднання.

Особливістю системи є наявність спеціального приводу, що дозволяє керувати конструкціями, не підключеними до джерела живлення. Зовнішній вигляд системи для керування воротами Hörmann наведено на рисунку 1.8.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		23

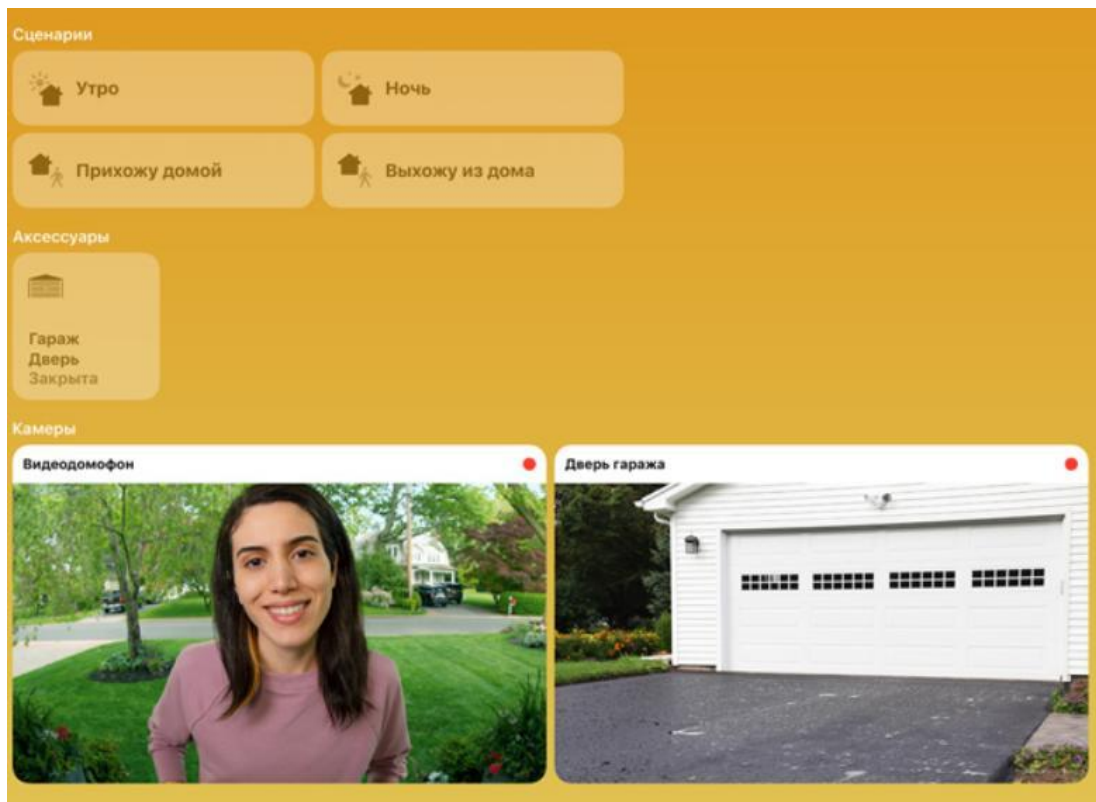


Рисунок 1.7 – Зовнішній вигляд інтерфейсу системи домашньої автоматизації HomeKit



Рисунок 1.8 – Зовнішній вигляд системи для керування гаражними воротами Hörmann

Система автоматизації воріт Alutech Levigatovиробництва міжнародного холдингу Alutech [25] надає можливість програмування рооти пристрою за допомогою однієї кдавіші і характеризується простотою керування. Зовнішній вигляд системи наведено на рисунку 1.9.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		24



Рисунок 1.9 – Зовнішній вигляд системи для керування гаражними воротами Alutech Levigato

Основними недоліками комерційних систем розумної автоматики для гаражних воріт є висока вартість самого обладнання, його монтажу, а також неможливість додавання власних функцій управління і безпеки.

1.6 Висновки. Постановка задачі

На основі проведеного дослідження функцій систем домашньої автоматизації, а також відомих комерційних систем розумної автоматики для гаражних воріт як складової систем домашньої автоматизації. Було зроблено висновок, що основним недоліком комерційних рішень є висока вартість самого обладнання, його монтажу, а також неможливість додавання власних функцій управління і безпеки.

В той же час використання платформи Arduino для проектування систем домашньої автоматизації надає важливі переваги, серед яких: можливість збору системи з окремих складових на зразок конструктора, широкі можливості в плані налаштування роботи системи домашньої автоматизації; можливість автономного функціонування при наявності власного контролера; низька вартість комплектуючих; наявність відкритого вихідного коду, що надає можливість безпосередньо користувачеві управляти механізмами системи домашньої

автоматизації; доступність та прозорість для користувача; універсальність і можливість реалізації будь-яких ідей.

Дослідивши основні архітектурні особливості апаратної платформи Arduino, а також особливості інтеграції сервоприводів в системи розумної автоматки для гаражних воріт, було визначено наступні задачі, які необхідно вирішити для проектування власного програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino:

- 1) Реалізувати зв'язку плат Arduino Uno та Ethernet Shield.
- 2) Протестувати коректність роботи зв'язки плат Arduino Uno та Ethernet Shield.
- 3) Підключили до Arduino Uno реле.
- 4) Помістити в корпус зв'язку плат Arduino Uno та Ethernet Shield і реле.
- 5) Підключити реле до воріт.
- 6) Написати базовий код прошивки для керування реле та створити відповідний інтерфейс.
- 7) Виконати прошивку Arduino Uno базовим кодом.
- 8) Написати модифіковану прошивку, що дозволить підвищити функції безпеки програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.
- 9) Протестувати коректність керування програмно-апаратним пристроєм віддаленого керування системою відкриття гаражних воріт на базі Arduino з різних пристроїв (смартфон, планшет тощо).

Реалізація вищенаведених кроків надасть можливість спроектувати власний програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino. Це дозволить реалізувати власний пристрій, в якому усунуто перераховані недоліки комерційних систем розумної автоматки для гаражних воріт.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		26

2 ПРОЕКТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ПРИСТРОЮ ВІДДАЛЕНОГО КЕРУВАННЯ СИСТЕМОЮ ВІДКРИТТЯ ГАРАЖНИХ ВОРІТ НА БАЗІ ARDUINO

З метою проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino потрібно здійснити вибір та обґрунтування вибору складових, а також програмного забезпечення, необхідних для реалізації проектованого програмно-апаратного пристрою.

2.1 Обґрунтування вибору плати Arduino Uno Rev 3

З метою проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino використано дві апаратні плати, які будуть детально досліджені нижче.

Перша плата – це плата розробки Arduino Uno [26], яка містить мікроконтролер, а друга – плата Ethernet Shield [27], яка підключає плату Uno до мережі Ethernet.

Для проектування було обрано плату Arduino Uno 3-ї ревізії, на що важливо звернути увагу, оскільки гнізда децю відрізняються в різних ревізіях плат Arduino. Зовнішній вигляд плати Arduino Uno Rev 3 наведено на рисунку 2.1. З зовнішнього вигляду плати Arduino Uno можна легко визначити плату третьої ревізії, оскільки кнопку скидання було переміщено з центральної правої сторони (як було на попередніх версіях) до верхньої лівої сторони (на платах третьої ревізії).

Основні характеристики плати Arduino Uno наведені в таблиці 2.1 [17].

Найважливішим пунктом, про який слід пам'ятати щодо Arduino Uno – це те, що це плата мікроконтролера, а не повністю робочий комп'ютер, такий як Raspberry Pi. Головна відмінність полягає в тому, що Arduino Uno не має можливості розміщення операційної системи і не може підтримувати будь-яку самостійну розробку програмного забезпечення, використовуючи лише плату.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		27

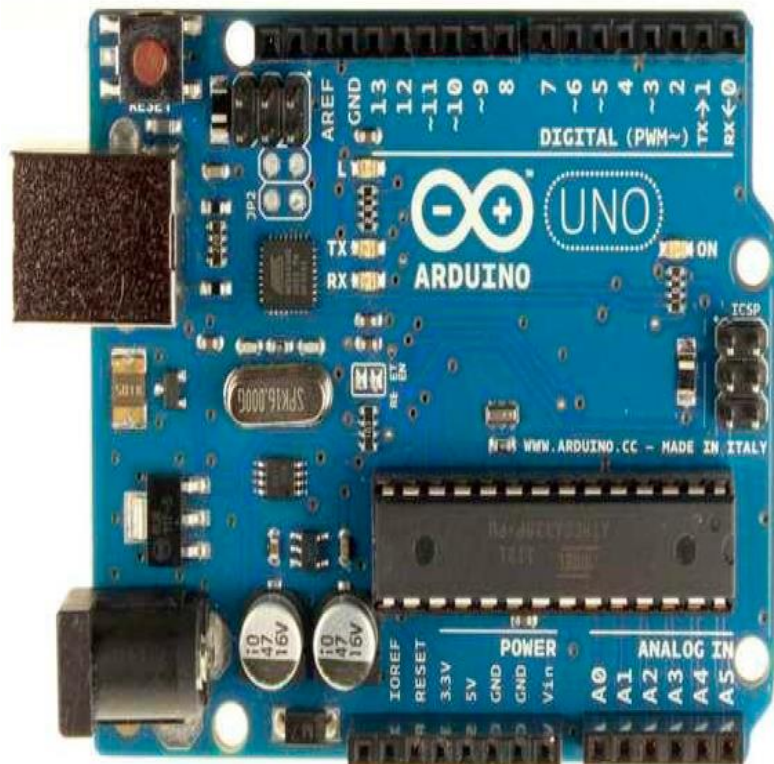


Рисунок 2.1 – Зовнішній вигляд плати Arduino Uno 3-ї ревізії

Таблиця 2.1 – Головні характеристики плати Arduino Uno

Складова	Характеристики	Примітка
Microcontroller	8-bit Atmel ATMEGA328P	28-pin DIP socket
Operational voltage	5V	Input range: 7–12V
Digital GPIO	14	6 capable of PWM
Analog IO	6	10-bit
Program memory	Flash 32KB,EEPROM 1KB	
Clock speed	16MHz	
USB	Type B socket	USB-based
Programmer	SPI,I2C	Software UART
Other	RTC,watchdog,interrupts	

Для програмування плати потрібно підключити її до зовнішнього комп'ютера. Це не робить Uno гіршим за Pi, він просто розроблений для іншого підходу – для управління вбудованими проектами, на відміну від Pi або Beaglebone Black. До проекту Arduino з відкритим кодом можна долучитись за

адресою <http://arduino.cc>, яка містить безліч посилань на інші сторінки, які є доволі інформативними.

Під час розробки потрібно перейти на цей сайт і ознайомитися з концепцією Arduino, оскільки це допоможе зрозуміти програмне забезпечення, що лежить в основі плат Arduino.

Наведені характеристики демонструють, що плата Arduino Uno Rev 3 є прийнятним варіантом для проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino, оскільки вона відповідає всім необхідним вимогам, що висувуються до проектування такого пристрою.

2.2 Обґрунтування вибору плати Ethernet Shield

При проектуванні програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino виникає необхідність підключення плати Arduino Uno до мережі Ethernet. З цією метою було вирішено використати плату Ethernet Shield [28]. На рисунку 2.2 показано зовнішній вигляд плати Ethernet Shield, яку було безпосередньо використано для встановлення дротового мережевого з'єднання між платою Arduino Uno та локальною мережею (LAN).

Слово “Shield” у назві пристрою стосується будь-якої з ряду плат, призначених для підключення безпосередньо поверх мікроконтролера Arduino, наприклад Arduino Uno. У ній міститься все обладнання, необхідне для встановлення дротового з'єднання між Arduino Uno та локальною мережею. Ця плата є старою версією Arduino Ethernet Shield і базується на чіпі Wiznet W5100.

Важливо зазначити, що хоча мікросхема W5100 містить 16 КБ для буферів пам'яті, вона не має жодних можливостей для зберігання програми, і вона повністю залежить від запрограмованих інструкцій Arduino Uno для того щоб правильно функціонувати. Це означає, що окрема бібліотека Ethernet повинна бути завантажена у флеш-пам'ять Arduino Uno, щоб все працювало належним чином.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						29
Зм.	Арк.	№докум.	Підпис	Дата		

Цей підхід, як правило, справедливий для більшості плат типу “Shield”, призначених для використання з мікроконтролерами серії Arduino [28].

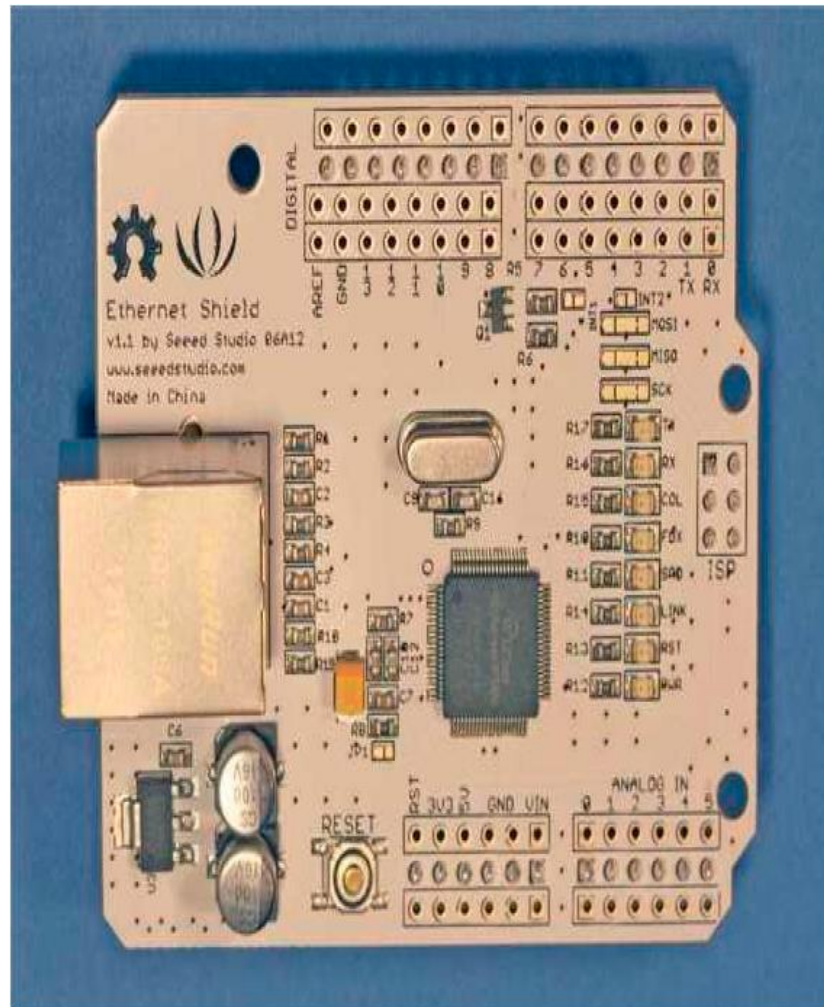


Рисунок 2.2 – Зовнішній вигляд плати Ethernet Shield

На рисунку 2.3 представлено блокову діаграму плати Ethernet Shield W5100, де відображено зв'язки між всіма компонентами, що складають вбудовані мережеві функції.

Таким чином, плата Ethernet Shield W5100 є прийнятним варіантом для підключення програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino до мережі Ethernet.

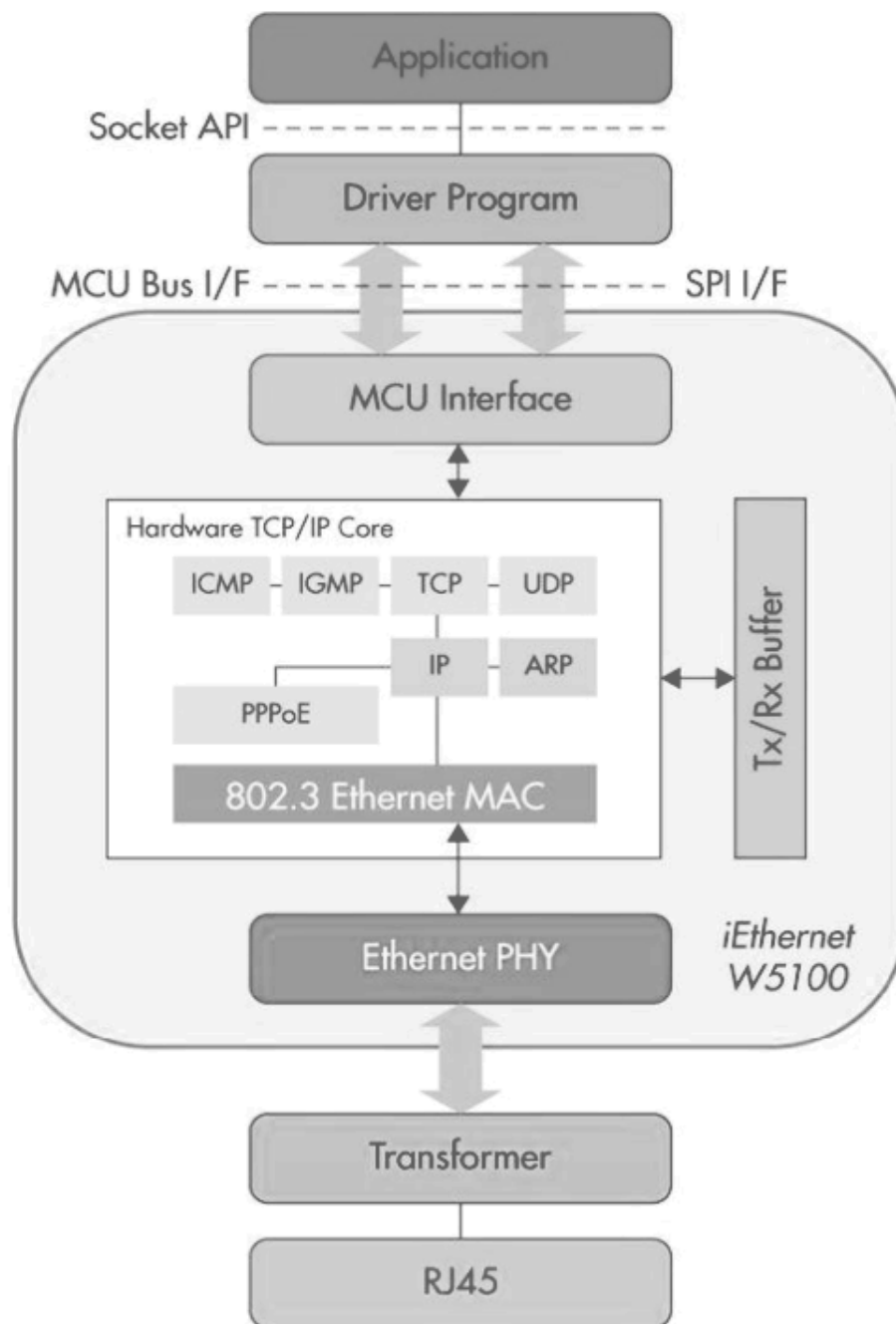


Рисунок 2.3 – Блокова діаграма мікросхеми Ethernet Shield W5100

2.3 Обґрунтування вибору програмного забезпечення для програмування плати Arduino Uno Rev 3

Ключовим програмним забезпеченням, яке потрібно для програмування Arduino Uno, є інтегроване середовище розробки (IDE). Uno IDE – це не Eclipse

IDE [29], але це повнофункціональний набір, розроблений спеціально для підтримки серії мікроконтролерів Arduino.

IDE можна безкоштовно завантажити з веб-сайту Arduino, який було надано раніше. Поточна IDE, яка буде використовуватись, має версію 1.05 [30], яка, ймовірно, зміниться в майбутньому, оскільки вдосконалення та оновлення постійно додаються людьми, які керують та підтримують проект Arduino. Скріншот екрану запуску Arduino v1.05 на ОС Windows зображено на рисунку 2.4.

Однією з особливостей є те, що існуюче обладнання Arduino завжди працюватиме на останній версії IDE. Жодного запланованого чи незапланованого старіння не передбачено.

Рекомендовано увімкнути пристрій Uno та підключити його до комп'ютера, на якому працює IDE, за допомогою стандартного кабелю USB. Працюватиме майже будь-яке джерело живлення, що використовує зовнішній роз'єм 2.1 мм із позитивним центральним контактом.

Слід зауважити, що напруга повинна бути від 7 до 12 В постійного струму. В проекті було використано надлишкове джерело живлення, що забезпечує 7,5 В при 1,5 А, що є більш ніж достатнім для цього проекту.

Комп'ютер повинен показати діалогове вікно про встановлення драйвера після підключення Uno до комп'ютера. Потрібно зачекати, поки драйвер не буде встановлений, перед запуском IDE.

IDE автоматично створив ескіз за замовчуванням, із назвою sketch_may04a, який, очевидно, що він містить дату запуску програми IDE.

На наступному кроці необхідно протестувати коректність роботи плати. Зазвичай з цією метою використовується порожній ескіз, а потім йому надається нова назва, що підходить для розроблюваної програми. Для цієї демонстрації не буде створений новий ескіз, а натомість буде завантажено попередньо збережений приклад Blink, який є стандартним і дозволяє продемонструвати роботу вбудованого світлодіода.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						32
Зм.	Арк.	№докум.	Підпис	Дата		

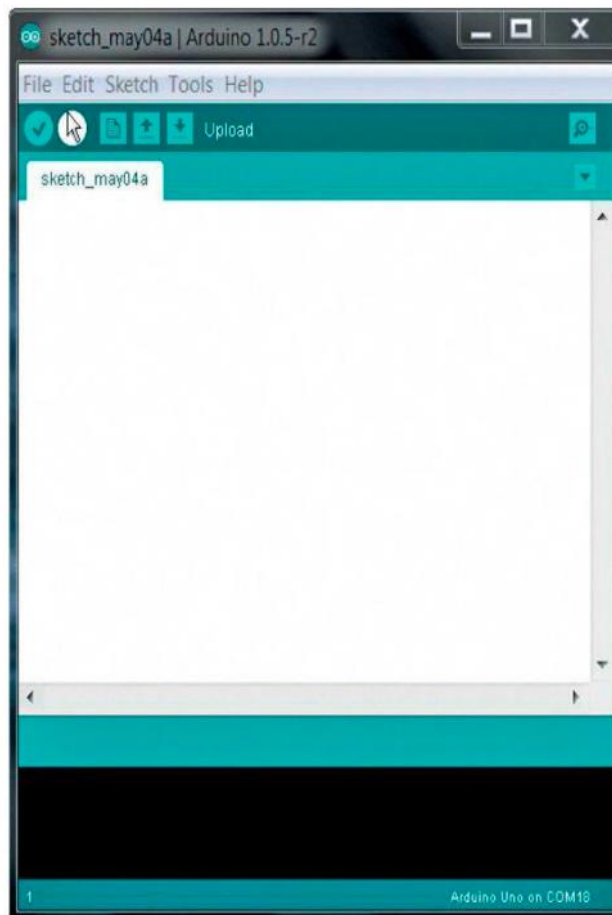



Рисунок 2.4 – Скріншот екрану запуску Arduino v1.05 на ОС Windows

Існує багато прикладів програм, які автоматично завантажуються в комп'ютер під час завантаження IDE. Програма, яку було відкрито, має назву “Blink” і завантажувалася дотримуючись наступної послідовності: вибрати Файл → Відкрити → Приклади → 01.Основи → Blink.

На рисунку 2.5 показана завантажена програма Blink, яка з'являється у власному вікні.

Потрібно звернути увагу, що оригінальне вікно для sketch_may04a все ще відкрите у фоновому режимі. Це робить роботу з декількома програмами дуже простою та зручною, оскільки все, що вам потрібно зробити, це вибрати потрібне вікно, щоб відновити розробку в цій програмі.

The image shows a screenshot of the Arduino IDE interface. The title bar reads "Blink | Arduino 1.0.5-r2". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for saving, opening, and other functions. The main text area displays the following code:

```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

  This example code is in the public domain.
  */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Рисунок 2.5 – Програма Blink

Цей ескіз має два методи: `setup` та `loop`. Метод `setup` завжди запускається спочатку, а потім – метод `loop`. Метод `setup` надає логічну назву "led" світлодіоду, прикріпленому до виводу 13 плати Uno. Він також робить виходом 13 вивід GPIO.

Метод `loop` – це вічний цикл, який по черзі включає світлодіод на одну секунду, а потім вимикає його на одну секунду.

Потрібно зауважити, що немає необхідності у приєднанні будь-яких пристроїв для цієї демонстрації, оскільки плата Uno вже має жовтий світлодіод, постійно підключений до виводу 13. Цей світлодіод можна побачити на рисунку 2.1, він позначений написом "L" і знаходиться трохи ліворуч і вище назви шовкографією ARDUINO.

Вибір стрілки, спрямованої праворуч, на панелі інструментів, показаній на рисунку 2.5, призведе до компіляції та завантаження програми в Uno.

Програма Blink розпочнеться негайно і триватиме нескінченно довго. Можна трохи заплутатися, оскільки світлодіод, ймовірно, вже блимав, перш ніж завантажити програму Blink. Це блимання показує, що плата працює за замовчуванням у режимі, який умикається, коли раніше не завантажувалася жодна програма.

Можна довести, що програма Blink працює належним чином, змінивши час затримки та спостерігаючи, що нова швидкість блимання відповідає введеному значенню.

Просто ввести нові значення часу затримки, скажімо, 2000 мс, що призведе до того, що світлодіод буде блимати з інтервалом дві секунди.

Щоб зібрати та завантажити програму потрібно натиснути стрілку вправо, і можна спостерігати як світлодіод повільно блимає кожні дві секунди.

2.4 Підключення плат керування Ethernet Shield та Arduino Uno та тестування їх спільної роботи

Розглянемо принципи використання плати Ethernet Shield спільно з Arduino Uno [28].

Першим кроком є відключення живлення Uno та USB-кабелів. Ніколи не слід намагатися прикріпити або зняти Ethernet Shield, поки Uno увімкнено, оскільки ви можете його пошкодити. Існує лише один спосіб вирівнювання плати типу Shield з входами Uno.

Потрібно ретельно вирівняти плати і міцно притиснути їх між собою. На рисунку 2.6 показано Ethernet Shield, прикріплений до Uno. Потрібно взяти до уваги, що роз'єм Ethernet Shield знаходиться на тій самій стороні, що і роз'єми живлення Uno та USB. Плата Shield ідентична за розміром платі Uno і повинна точно покривати його.

					КВРКІ.170294.17.02.24 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		35



Рисунок 2.6 – Ethernet Shield прикріплений до Arduino Uno

Після цього потрібно підключити кабель живлення та USB Uno, а також підключити патч-кабель Ethernet від плати Shield до маршрутизатора, концентратора або комутатора вашої мережі, залежно від того, як ви налаштували свою локальну мережу.

У цьому випадку було виконано підключення до перемикача 10/100. Після цього можна побачити на екрані чотири-п'ять зелених світлодіодів, які вказують, що живлення подається на плату, а мережева активність виявляється за допомогою патч-кабелю. На рисунку 2.7 показані ці активні світлодіоди.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		36

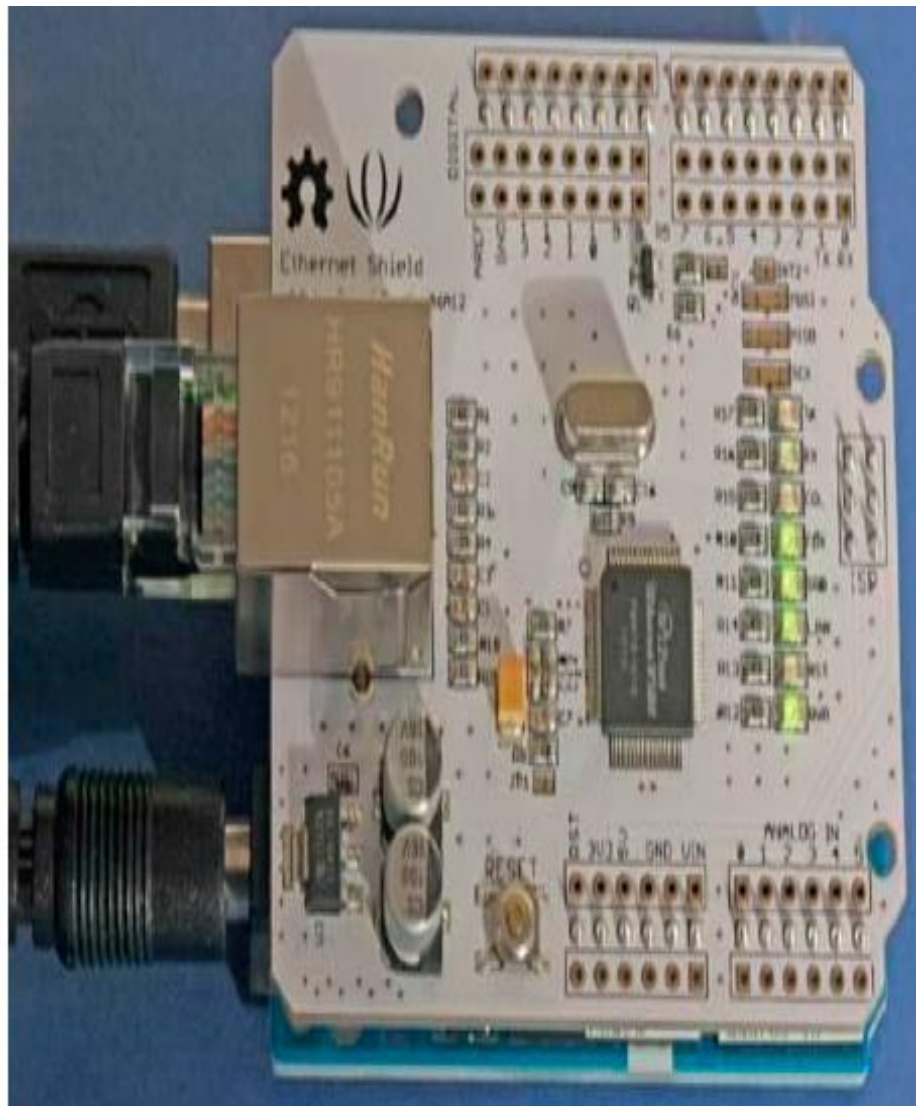


Рисунок 2.7 – Ethernet Shield з світлодіодами, що вказують на коректну роботу пристрою

Якщо немає індикації цих світлодіодів, негайно відключіть живлення від Uno і перевірте, як Ethernet Shield і Uno з'єднані між собою. Найімовірніша причина полягає в тому, що піни були зсунуті на одну або навіть дві позиції. Необхідно вирівняти плату Shield і прикріпити її до Uno.

Бібліотека Ethernet, до якої потрібно буде отримати доступ, доступна з меню меню IDE: Відкрити → Приклади → Меню Ethernet. Усі поточні пункти меню бібліотеки Ethernet детально описані в таблиці 2.2.

Зм.	Арк.	№докум.	Підпис	Дата

Таблиця 2.2 – Опис бібліотеки Ethernet

Ім'я програми	Опис
ChatServer	Запускає простий сервер чату
ChatClient	Запускає простий клієнт для взаємодії з сервером чату
WebClient	Робить HTTP-запит
WebServer	Розміщує просту HTML-сторінку, що відображає шість аналогових значень
PachubeClient	Використовує Xively.com, безкоштовний сайт реєстрації даних
PachubeClientString	Надсилає дані на Xively.com.
BarometricPressureWebServer	Виводить значення з барометричного датчика тиску
UDPSendReceiveString	Надсилає та отримує текстові записи через UDP
LidpNtpClient	Query сервер мережевого часу (NTP) за допомогою UDP
DnsWebClient	Веб-клієнт на базі DNS і DHCP.
DhcpChatClient	Простий сервер чату DHCP.
DhcpAddressPrinter	Отримує IP-адресу через DHCP і виводить її
TelnetClient	Простий Telnet клієнт

Потрібно відкрити програму DhcpAddressPrinter і завантажити її в Uno. Після завантаження необхідно вибрати піктограму підзорної труби біля верхнього правого кута IDE, щоб відкрити послідовний термінал. Після цього можна побачити відображену IP-адресу Uno, як показано на рисунку 2.8.

Зм.	Арк.	№докум.	Підпис	Дата

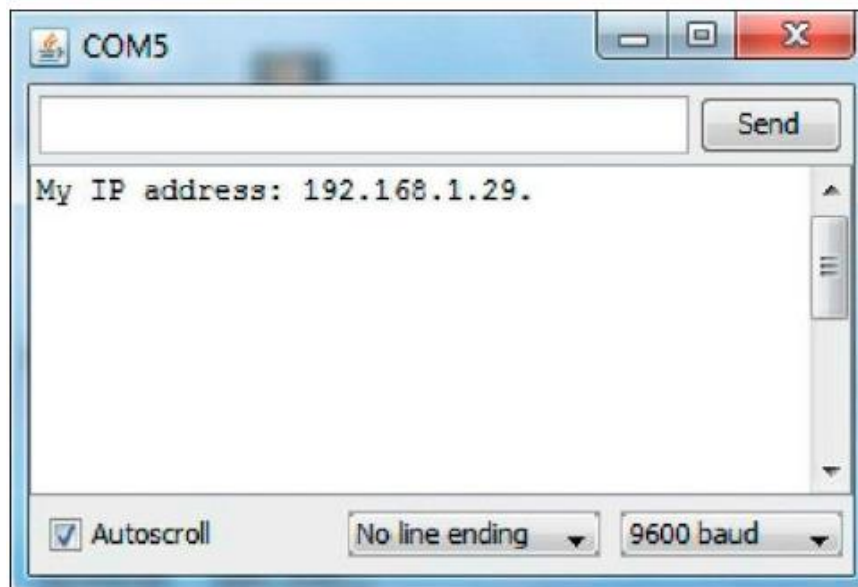


Рисунок 2.8 – Відображення послідовного терміналу з IP адресою Arduino

Мережевий маршрутизатор призначив IP-адресу 192.18.1.29 для Ethernet Shield. Однак адреса, безсумнівно, може бути іншою. Цей адрес знадобиться для наступної програми. Потім потрібно завантаж програму WebServer, що знаходиться в бібліотеці Ethernet, в Uno.

Ця програма створює активний сервер, який відповідає на запити HTTP за IP-адресою, яку визначили за допомогою попередньої програми, і повертає вихідні дані з шести аналогових входів введення Uno.

WebServer використовує HTTP-порт 80 за замовчуванням, тому потрібно буде лише ввести IP-адресу на віддаленому мережевому комп'ютері, щоб побачити результати. Однак необхідно зробити наступні зміни в WebServer, щоб він працював із попередньо призначеною IP-адресою Uno.

Необхідно звернути увагу, що номери IP відокремлюються (розділяються) комами, а не крапками, як зазвичай.

Для цього та всіх інших програм Arduino Ethernet у цьому проекті використовується фіктивна MAC-адреса. Є дві причин, по-перше, не має різниці, якою є фактична адреса MAC, якщо вона є унікальною в мережі.

Друга причина полягає в тому, що в комплект Ethernet Shield не входила жодна наклейка, що відображає фактичну MAC-адресу мікропрограми, призначену чіпу.

Ця ситуація справедлива для більшості старих версій Ethernet Shields. У такому випадку все, що потрібно зробити, це змінити попередньо встановлену MAC-адресу, щоб вона відповідала адресі наклейки. Це також має спрацювати, якщо не будуть внесені зміни, але вважається, що є гарною практикою завжди збігати адреси MAC, якщо це можливо.

Тепер слід відкрити браузер на мережевому комп'ютері та ввести IP-адресу Uno, яка у цьому випадку становила 192.168.1.29. Вводити префікс “http: //” абсолютно необов'язково, оскільки всі сучасні браузери вважають, що це протокол за замовчуванням. Рисунок 2.9 показує вивід на MacBook Pro, коли було перейдено за цим адресом.

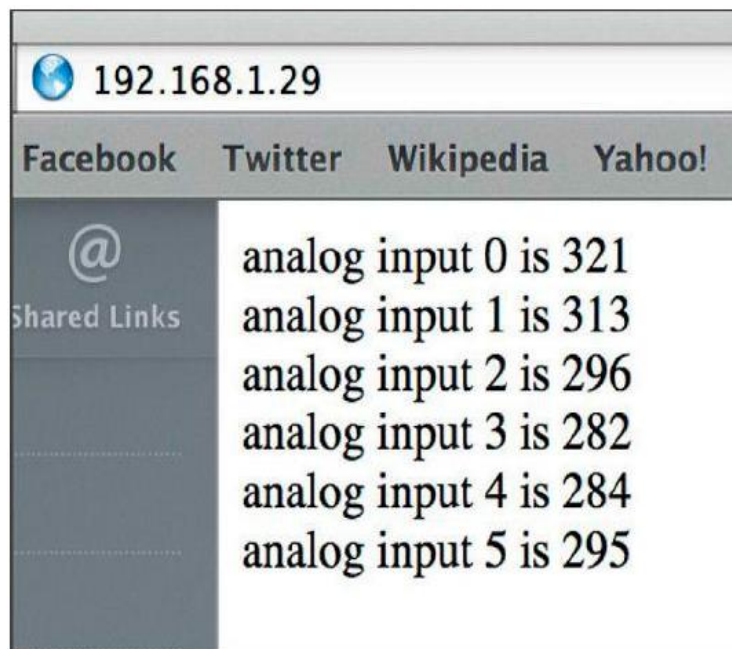


Рисунок 2.9 – Зовнішній вигляд інтерфейсу програми WebServer, відкритої у віддаленому браузері

На рисунку можна побачити цифри в діапазоні від найнижчого 282 до максимального 321. Вони представляють напругу, що “плаває” на шести не пов'язаних аналогових вхідних каналах Uno.

Кожен канал має роздільну здатність 10 біт і має повномасштабну вхідну напругу 5.0 В, що відповідає робочій напрузі Uno. Це означає, що число 1023 буде представляти 5.0 В для певного аналогового каналу. Отже, цифри, що

відповідають приблизно 300, представляють приблизний рівень 1.47 В, який визначається простим пропорційним розрахунком:

Плаваюча вхідна напруга обчислюється наступним чином: $U = 5.0 * (300/1023) = 1.47$ В.

Як можна побачити з рисунку 2.10, була припаяна перемичка між землею та аналоговим входом А0.

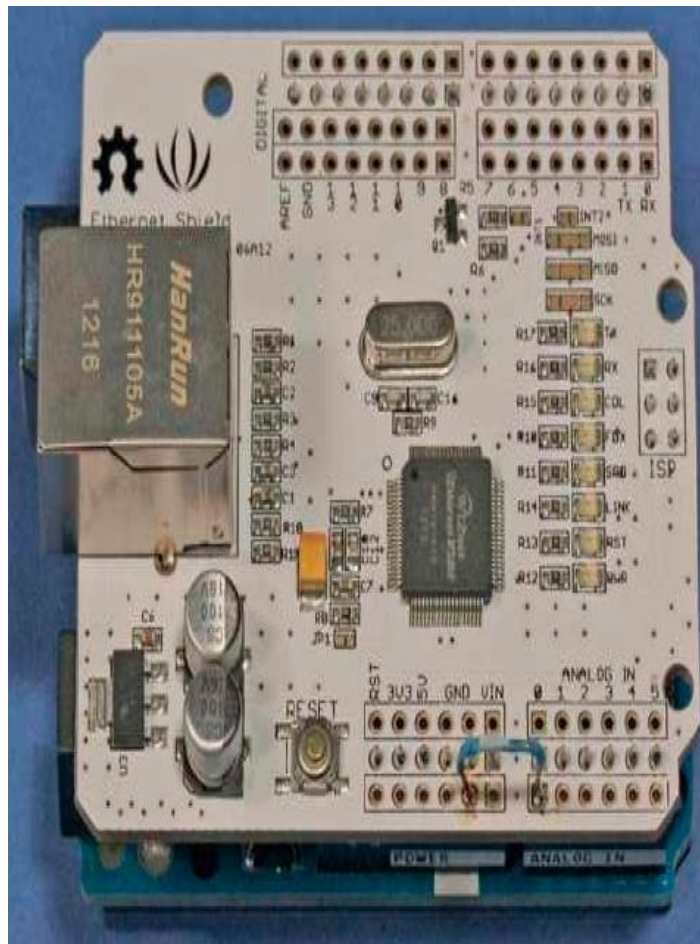


Рисунок 2.10 – Перемичка, впаяна між землею та аналоговим входом А0

Це було зроблено з метою підтвердження, що аналогові входи працюють належним чином, оскільки очікувалось побачити рівень 0В, коли наступного разу було відкрито браузер браузер на сторінку WebServer. Про коректну роботу плати свідчить рисунок 2.11, на якому зображений WebServer, що працює із заземленим входом А0.

```
analog input 0 is 0
analog input 1 is 89
analog input 2 is 149
analog input 3 is 174
analog input 4 is 207
analog input 5 is 198
```

Рисунок 2.11 – Виведення WebServer при заземленому контакті A0

Таким чином, можна побачити, що не тільки вхід A0 має значення 0, але всі інші набагато нижчі, ніж були раніше, коли до входів нічого не було підключено.

Як висновок, що на входах повинні бути дуже високі комплексні опори, на які легко впливати блукаючі та прилеглі електричні поля. Це також пояснює, чому, при спробі виміряти напруги розімкнутого струму на аналогових входах за допомогою стандартного мультиметра, реєструвались набагато нижчі напруги, ніж було розраховувано.

Мультиметр з помірним імпедансом просто завантажував входи високого імпедансу і показував напруги, які були результатом шунтування мультиметра з високим імпедансом. Це хороший приклад визнання того, коли випробувальне обладнання може мати незрозумілий вплив на пристрій, що тестується (DUT).

DUT – це стандартний опис електроніки для будь-якого випробовуваного електричного / електронного обладнання. Цей можливий ефект слід завжди брати під час тестування будь-якого пристрою.

Продемонстроване підключення Uno до мережі та перевірка коректності роботи. Для цього потрібно задіяти програму, яка прийме введені користувачем дані браузера і змусить Uno виконати бажану операцію. Яка покаже основні поняття, що стосуються управління мікропроцесором через мережеве з'єднання.

Ця програма також стане основою для спрощеного проекту відкриття гаражних воріт. Для цієї початкової демонстрації буде запалений світлодіодний індикатор лише тоді, коли встановлено прапорець. Перед компіляцією коду слід

виконати просту настройку обладнання. Це передбачає послідовне підключення анода світлодіода до виводу 2 Uno та резистора 220 Ом від катода світлодіода до землі. На рисунку 2.12 показано фізичне налаштування. Потрібно звернути увагу, що дроти в цьому випадку були припаяні безпосередньо до контактів Ethernet Shields, які з'єднуються з контактом 2 та одним із контактів заземлення. Також цю зв'язку плат було підключено до джерела живлення 5 В, яке буде використано далі.

Код починається з двох операторів `include`, які посилаються на всі залежності, необхідні для роботи цієї програми. Це SPI та базові бібліотеки Ethernet. Далі йде “фіктивна” MAC-адреса, яку можна використовувати або ввести власну.

Наступна стрічка ініціює IP-адресу Uno. Її потрібно ввести, щоб відповідати реальній IP-адресі, інакше неможливо встановити підключення Ethernet, як вже говорилося у попередньому прикладі.

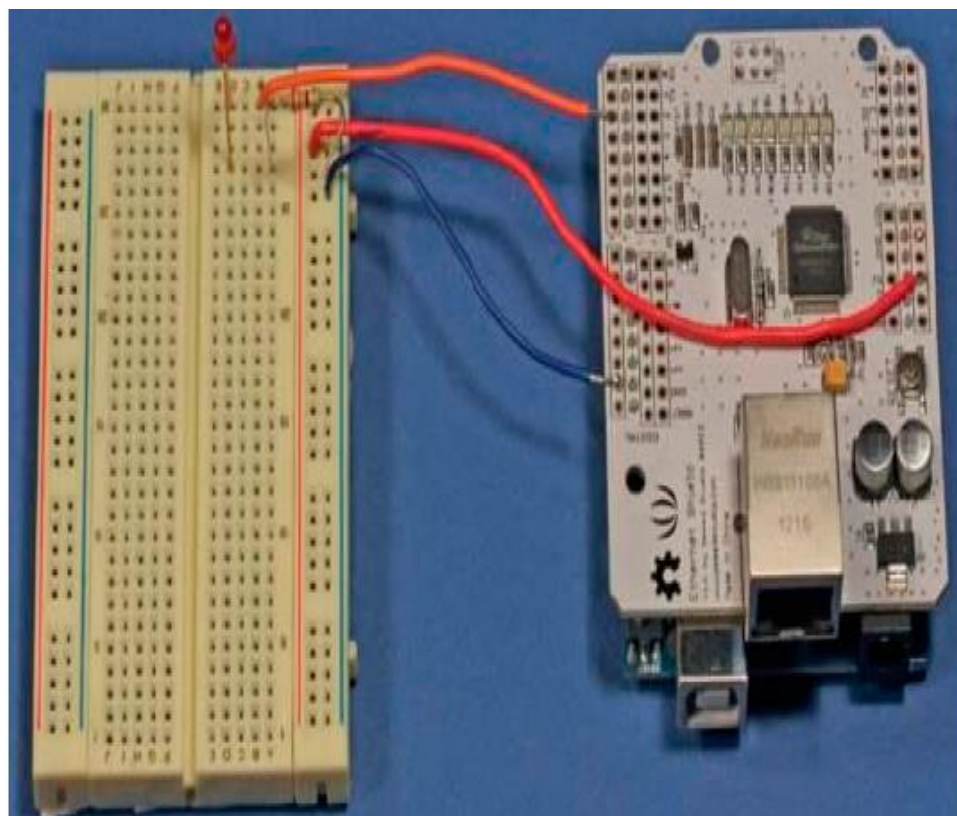


Рисунок 2.12 – Компоновка з світлодіодом для перевірки плати

Далі – два стандартні методи Arduino з назвою `setup` та `loop`. Цей конкретний спосіб налаштування виконує такі ініціалізації:

- 1) ініціювання об'єкта підключення Ethernet;
- 2) старт прослуховування запитів клієнтів;
- 3) запуск послідовного терміналу зі швидкістю 9600 бод;
- 4) встановлення 2 піну GPIO Uno як вихід.

Метод `loop` слідує за налаштуванням, окрім цього він більш об'ємний, ніж метод `loop` попереднього прикладу.

Основною причиною цього є те, що Arduino Uno не використовує операційну систему і, отже, не може налаштувати та підтримувати файлову систему.

Це означає, що код, який взаємодіє з віддаленим браузером (клієнтом), повинен динамічно створюватися кожного разу, коли запускається програма веб-сервера. Мета всіх інструкцій `client.println`, що містяться в методі `loop`.

Цю необхідність слід порівняти з тим, як функціонує веб-служба Raspberry Pi, Pi має повну файлову систему Linux, що означає, що клієнтський код може зберігатися у заздалегідь визначеному каталозі, з якого користувачі можуть отримати до нього доступ. Зазвичай він знаходиться в каталозі `/etc/www`, а клієнтський файл, як правило, називається `index.htm` або `index.html` для протоколу HTTP з використанням порту 80.

Перший оператор у методі `loop` – `EthernetClient client = server.available ();` який присвоює ненульове значення посиланню на клієнта, якщо виявлено запит HTTP. Далі будуть виконані всі оператори `println`, в результаті чого веб-сторінка відобразатиметься на стороні клієнта.

Сторінка, показана на рисунку 2.13, з'являється, коли клієнт, переглядаючи віддалений комп'ютер мережі з'єднується з програмою веб-сервера Uno.

Тепер метод `loop` буде циклічно чекати, поки не встановлено прапорця, вказуючи тим самим, що користувач хоче увімкнути підключений світлодіод. Перевірка стану прапорця – це функція третього методу, що міститься в підпрограмі `ProcessCheckbox`.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						44
Зм.	Арк.	№докум.	Підпис	Дата		

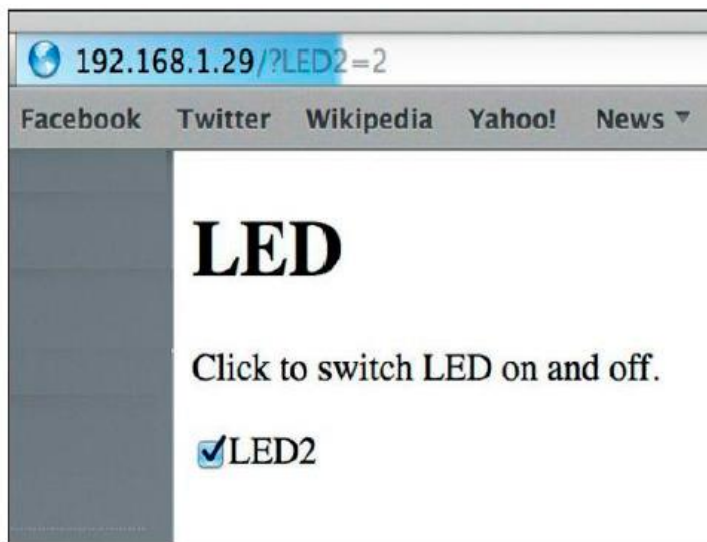


Рисунок 2.13 – З’єднання клієнта з веб-сервером Arduino Uno

Тепер метод `loop` буде циклічно чекати, поки не встановлено прапорець, вказуючи тим самим, що користувач хоче увімкнути підключений світлодіод. Перевірка стану прапорця – це функція третього методу, що міститься в підпрограмі `ProcessCheckbox`.

Цей метод приймає аргумент клієнта і може призвести до того, що світлодіод увімкнеться, якщо він раніше був вимкнений, або вимкне його, якщо він раніше вмикався. Стан світлодіода, незалежно від того, увімкнено чи вимкнено, зберігається у програмі, що має відповідну назву `LED_status`. Якщо прочитати код методу `ProcessCheckbox`, можна помітити, що HTML-виклик браузера абсолютно однаковий, незалежно від того, вмикаєте чи вимикаєте світлодіод. Цей виклик просто перемикає прапорець у прапорці і не має нічого спільного з фактичним увімкненням та вимкненням світлодіода.

Це все обробляє плата `Arduino Uno` у підпрограмі `WebServer`. Було запущено програму і можна спостерігти, як світлодіод вмикається та вимикається, як очікувалося, коли було встановлено прапорець. Наступним кроком є модифікація цієї прошивки для управління реле, яке в свою чергу управляє гаражними воротами.

2.5 Висновки

В другому розділі було проведено вибір складових для проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino, а також обґрунтовано цей вибір. Також було обрано програмне забезпечення для програмування плати Arduino Uno Rev 3.

Після цього було здійснено підключення плат керування Ethernet Shield та Arduino Uno та протестовано коректність їх спільної роботи.

Таким чином, наступним етапом роботи є програмно-апаратна реалізація та тестування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

					КВРКІ.170294.17.02.24 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		46

3 ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ПРИСТРОЮ ВІДДАЛЕНОГО КЕРУВАННЯ СИСТЕМОЮ ВІДКРИТТЯ ГАРАЖНИХ ВОРІТ НА БАЗІ ARDUINO

3.1 Модифікація комутатора панелі керування сервоприводом

Розглянемо процес проектування базового контролера з дистанційним управлінням для розроблюваного програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino. Базовий контролер з дистанційним управлінням призначений для роботи з існуючим типовим пристроєм для відкривання воріт [31], що показаний на рисунку 3.1.



Рисунок 3.1 – Сервопривід, використаний при проектуванні програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

Цей пристрій досить типовий для приводів гаражних дверей, які використовуються для житлового обслуговування. Його вмикає натискання настінної кнопки або натискання кнопки на передавачі радіоуправління, який прикріплений всередині автомобілю.

Суть полягає в тому, щоб паралельно підключити два дроти до тих проводів, які в даний час підключені до настінної кнопки. Ці два нові дроти будуть підключені до набору розімкнутих контактів реле, так що коли на реле подається напруга від $U_{по}$, він замикає контакти. Це відбувається аналогічно, як при натисканні настінної кнопки приводиться у дію механізм відкривання/закривання.

Реле необхідне, оскільки на проводах, що йдуть до кнопки, є потенціал 16 В постійного струму [32], що набагато перевищує характеристики напруги $U_{по}$ ГРІО. На рисунку 3.2 подано збільшене зображення підключення клем на комутаторі панелі керування сервоприводом, де були прикріплені паралельні кнопкам дроти. Це два крайні ліві дроти, показані на рисунку.



Рисунок 3.2 – Підключення клем на комутаторі панелі керування сервоприводом

З рисунку 3.1 можна побачити два додані нові дроти, що знаходяться з лівого боку приводу. З цією метою було використано одножильний,

					КВРКІ.170294.17.02.24 ПЗ	Арк
						48
Зм.	Арк.	№докум.	Підпис	Дата		

калібрувальний дріт розрізу 20 для усіх підключень, оскільки це такий самий тип, який був встановлений на настінній кнопці.

3.2 Створення прототипу програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

На наступному етапі створимо прототип програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino. З цією метою необхідно розмістити на макетній дошці реле, зв'язку плат Arduino Uno та Ethernet Shield та помістити їх в пластиковий корпус.

Базова програма управління світлодіодами була модифікована таким чином, що вона буде керувати транзистором, який, у свою чергу, буде керувати реле, до якого підключені дроти контакту гаражних воріт.

Електричну схему підключення реле зображено на рисунку 3.3. Для перемикання реле можна використовувати практично будь-який звичайний комутаційний транзистор NPN. У цій схемі був використаний транзистор у корпусі 2N3904 [33].

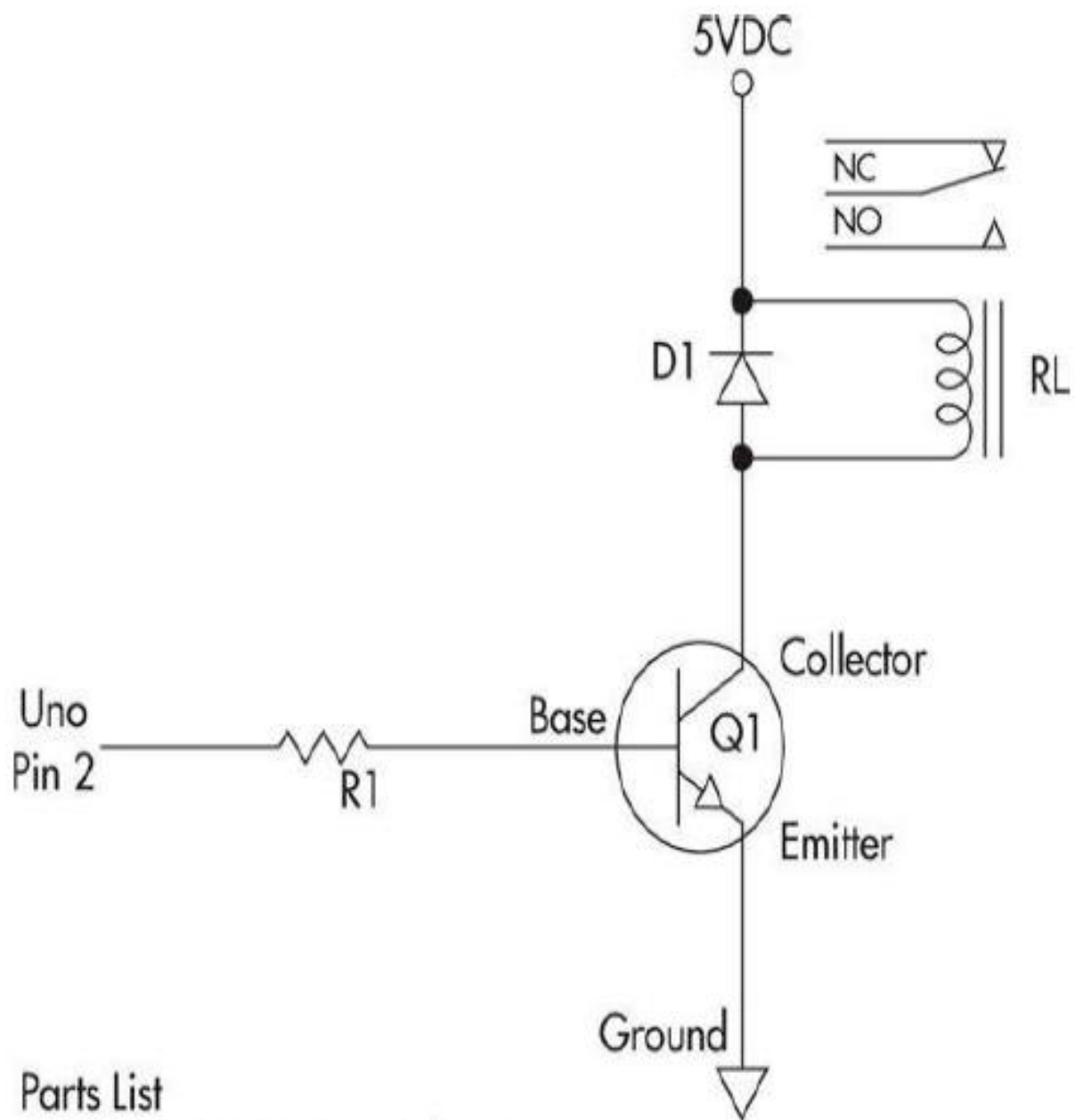
Схема перемикання реле була підключена до макетної плати [34] для зручності та спрощення модифікації при необхідності. Макет і плата Uno були встановлені в пластиковому корпусі [35] біля гаражних воріт.

На рисунку 3.4 показано плату Uno та реле, встановлену у корпусі.

Також було проведено кабель живлення і кабель Ethernet разом із парою дротів управління, підключеними до механізму відкривання гаражних воріт безпосередньо до корпусу.

Можна побачити всі з'єднання в корпусі, який був встановлений на стіні гаража. В додатку Г наведено початковий та модифікований коди програми для управління програмно-апаратним пристроєм віддаленого керування системою відкриття гаражних воріт на базі Arduino.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		49



Parts List

Q1- 2N3904 NPN switching transistor

R1- 560 ohm 1/4 watt resistor

D1- 1N4148 kickback diode

RL- 5V relay

Рисунок 3.3 – Електрична схема підключення реле

Зм.	Арк.	№докум.	Підпис	Дата

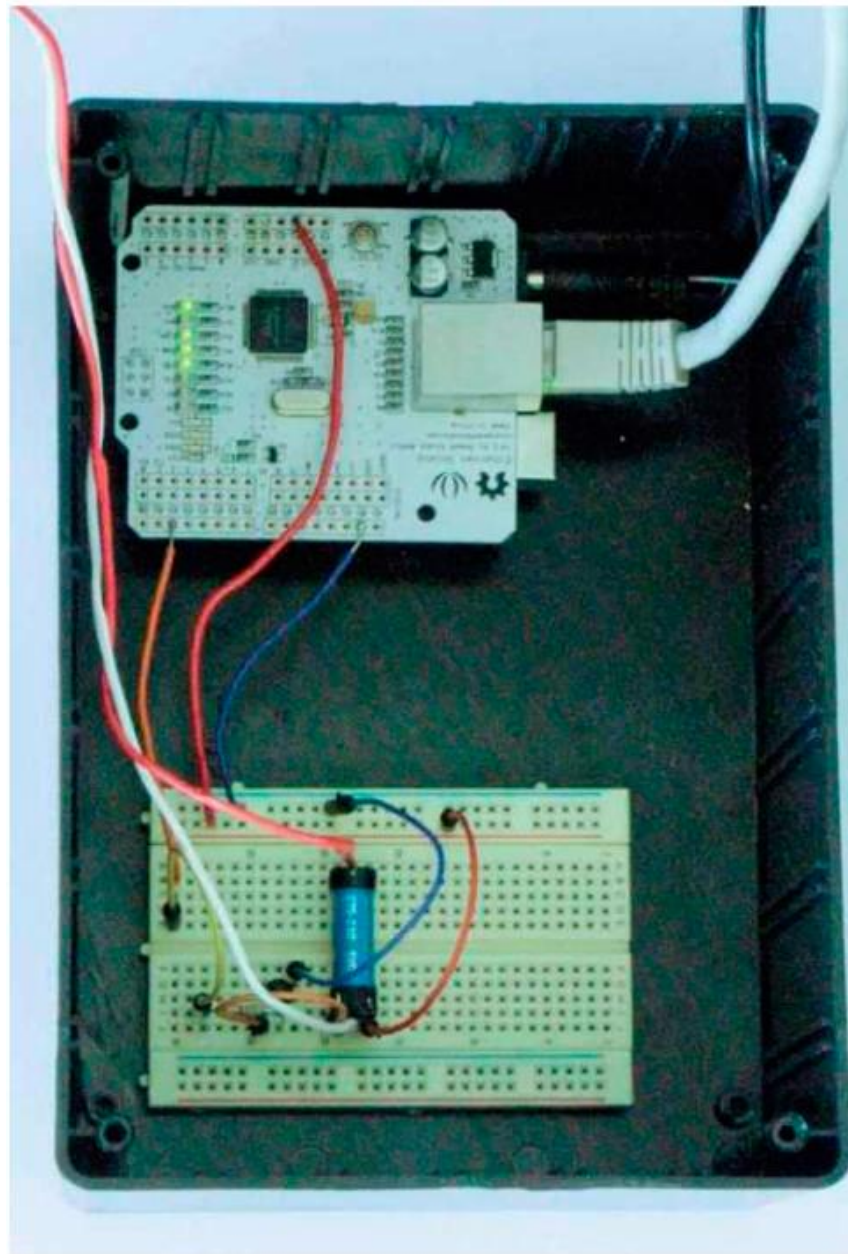


Рисунок 3.4 – Плата Arduino Uno rev 3 та реле, розміщені у пластиковому корпусі

На рисунку 3.5 відображено модифікований інтерфейс керування гаражними воротами, відкритий у браузері. Для забезпечення можливості дистанційного замикання реле на одну секунду кожного разу, коли встановлено прапорець DOOR, також було внесено додаткові корективи в код. Для управління гаражними воротами необхідно натиснути на елемент CheckBox DOOR.

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ.170294.17.02.24 ПЗ

Арк

51

Garage Door Opener

Click to activate door.

DOOR

Рисунок 3.5 – Інтерфейс керування гаражними воротами, відкритий у браузері

Установивши прапорець, можна відкривати або закривати гаражні ворота, залежно від їх попереднього положення.

Ця програма жодним чином не порушує властиві функції безпеки, що могли вплинути на гаражні ворота, наприклад у випадку наявності перешкоди. Гаражні ворота припинять рух і змінять його напрямок, якщо зустрінуть перешкоду при закритті. Крім того, захисні інфрачервоні датчики внизу гаражних воріт також функціонують.

Окрім цього, можна керувати гаражними дверима за допомогою смартфона. Для цього потрібно ввести IP-адресу Uno у браузер телефону і підключитися до веб-сервера. На рисунку 3.6 наведено зовнішній вигляд вікна браузера, відкритого на смартфоні.

При цьому гаражні ворота працюють аналогічно, як і при використанні браузера, відкритого з ноутбука. Звичайно, в цьому варіанті не існує жодної додаткової безпеки, крім ключа доступу WPA, який використовується в безпечному з'єднанні Wi-Fi локальної мережі.

					КВРКІ.170294.17.02.24 ПЗ	Анк
Зм.	Арк.	№докум.	Підпис	Дата		52

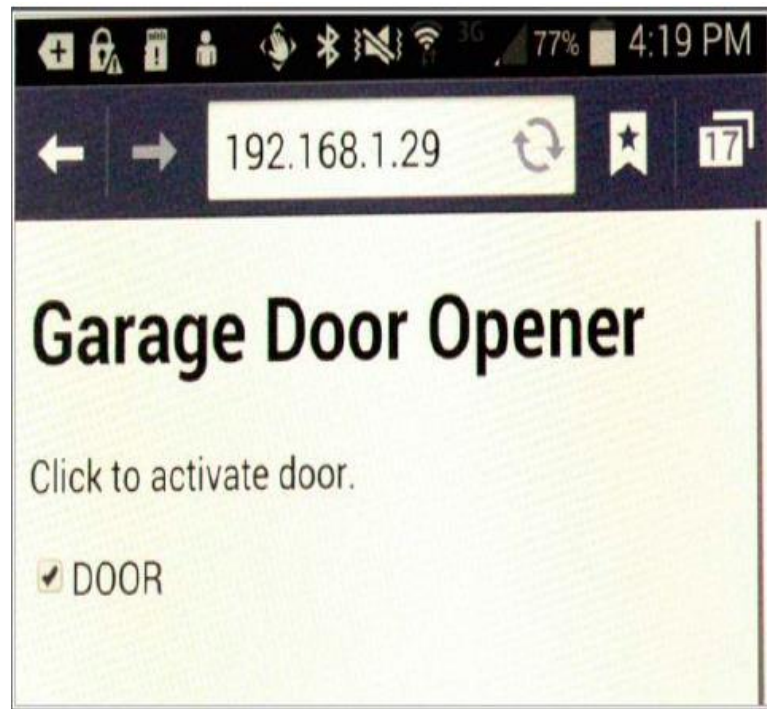


Рисунок 3.6 – Зовнішній вигляд вікна браузера, відкритого на смартфоні, підключеного до веб-сервера програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

Проте WPA є достатнім для запобігання несанкціонованому відкриттю гаражних воріт. Безумовно, заборонено використовувати це програмне забезпечення для гаражних воріт із відкритим з'єднанням Wi-Fi.

На наступному кроці внесемо ще одну модифікацію та залучимо ще одне IDE, що буде використане разом із IDE Arduino з метою створення програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

3.3 Використання середовища Visual Studio 2019 IDE для написання прошивки плати Arduino

Microsoft Visual Studio IDE [36] є надзвичайно корисним для розробки програмного забезпечення та використання з платою Arduino. Використання цієї IDE дає можливість отримати доступ до широкого спектру засобів програмування

та бібліотек, яких немає в більш обмежених бібліотеках Arduino. Для розробки цього програмного забезпечення знадобляться три програмні пакети:

- 1) Visual Studio 2019;
- 2) Arduino версії 1.05;
- 3) MegunoLink Pro [37].

Видання Microsoft Visual Studio 2019 (VS2019) буде використано для розробки програмного забезпечення для програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Це IDE доступне для завантаження безкоштовно для некомерційних цілей на веб-сайті Microsoft [36].

Це програма для Windows, і вона досить велика, тому завантаження займе певний час. Після встановлення доведеться зареєструватися, щоб отримати безкоштовний ключ продукту. В іншому випадку термін використання програмного забезпечення закінчується через 30 днів. Використання програмного забезпечення є безкоштовним лише у тому випадку, якщо виконується валідація, що користувач використовує його лише для особистих некомерційних цілей.

На рисунку 3.7 показано зовнішній вигляд середовища VS2019 без завантажених проектів.

Для сумісності програмного забезпечення з IDE Arduino потрібно врахувати, що це повинна бути версія 1.05 або пізніша, оскільки попередні версії можуть мати проблеми із сумісністю з VS2019.

Для реалізації програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino необхідно використати ще одне програмне забезпечення, що називається MegunoLink Pro [37] і слугує мостом між середовищем VS2019 та Arduino. Без використання програмного забезпечення MegunoLink Pro неможливо завантажити будь-який скомпільований проект VS2019 на плату Arduino. Для MegunoLink Pro визначено безкоштовний семиденний пробний період. Користувач має придбати ліцензію, якщо хочете користуватися нею після закінчення пробного періоду. Плата за ліцензію для особистого користування є цілком розумною, при цьому вона надає додаткову

розширену функціональність, дозволяючи використовувати VS2019 з платами Arduino, а також надаючи професійні графічні інтерфейси (GUI).

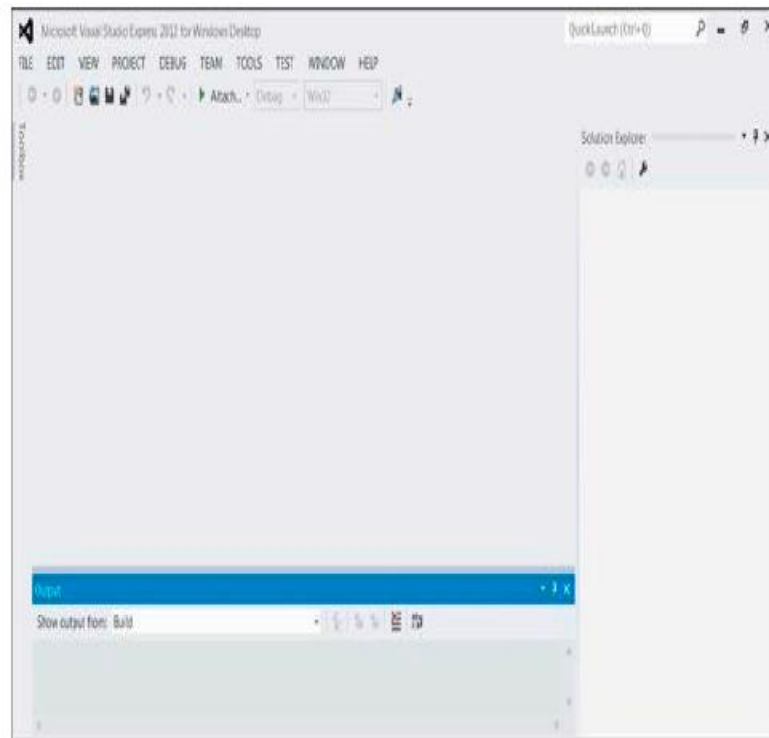


Рисунок 3.7 – Зовнішній вигляд середовища Visual Studio 2019 IDE

Потрібно дотримуватись інструкцій під час встановлення з веб-сайту, щоб спочатку налаштувати програмне забезпечення MegunoLink Pro. Коли з'явиться піктограма шестерні на панелі інструментів MegunoLink Pro, буде показано екран встановлення інструментів, як показано на рисунку 3.8. Потрібно вибрати налаштування Arduino Build Tool для Visual Studio 2019, щоб встановити програмний модуль у VS2019, який дозволяє завантажувати зібрані програми на плати Arduino.

Насамперед потрібно ввести розташування каталогу для двійкового виконуваного файлу `arduino.exe`, що може дещо відрізнятись залежно від параметрів, вибраних під час встановлення програмного забезпечення Arduino.

У цьому випадку місце розташування було визначено як “C: \ Program Files \ Arduino”, як можна бачити на рисунку 3.8.

Поточний послідовний порт, що підключає Arduino Uno до комп'ютера, також повинен бути вибраний у модулі візуалізатора MegunoLink Pro Connection Manager.

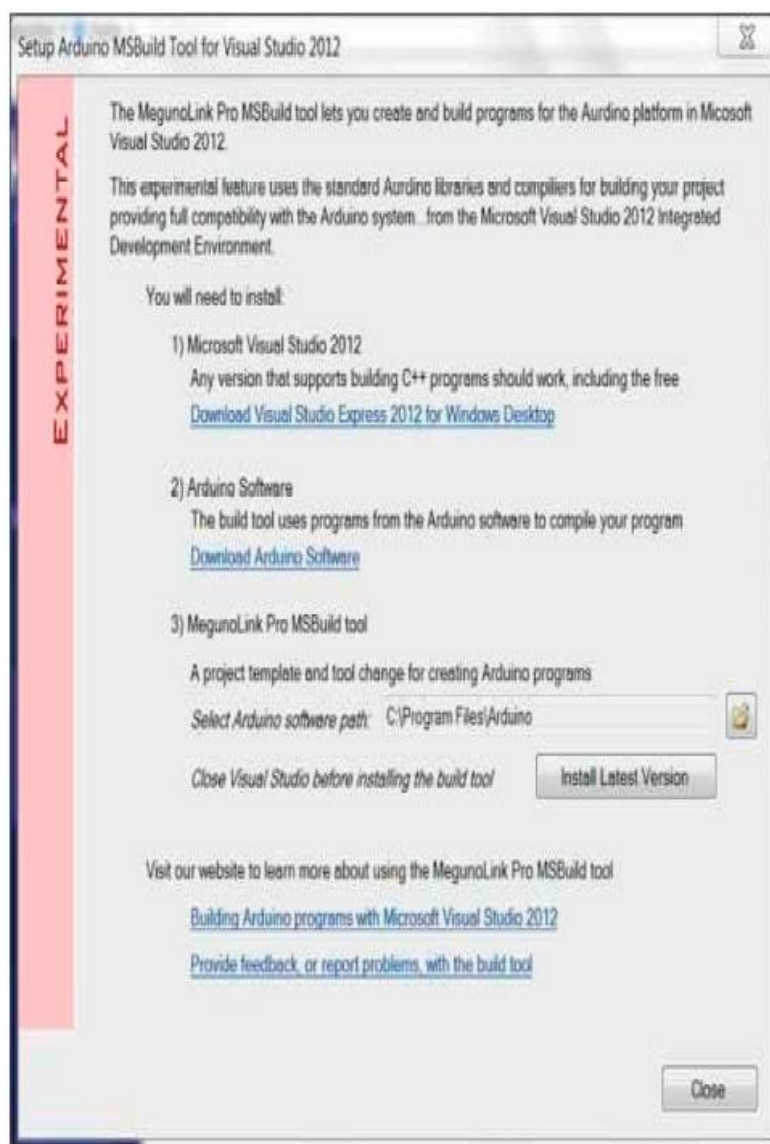


Рисунок 3.8 – Інтерфейс майстру з встановлення MegunoLink Pro

Arduino Uno повинна бути увімкнена і підключена до комп'ютера за допомогою кабелю USB, щоб послідовний порт розпізнавався диспетчером підключень. На рисунку 3.9 показано диспетчер підключень, який встановлює підключення через COM46 на 9600 бод.

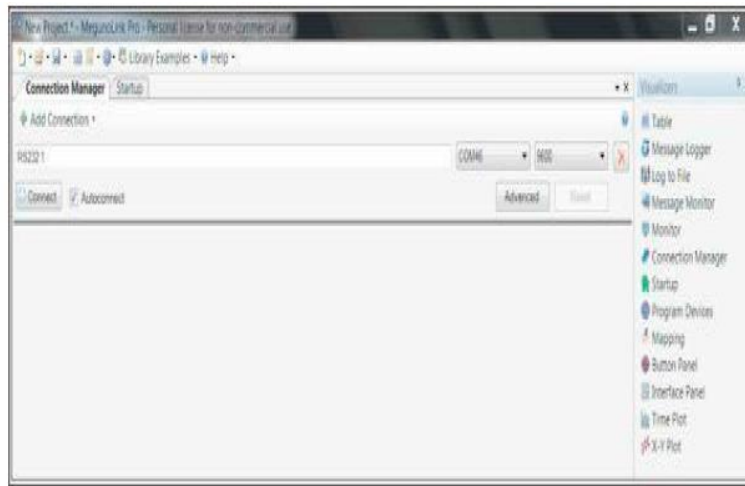


Рисунок 3.9 – Вікно диспетчера підключень

Виконання всіх попередніх кроків повинно дозволити програмам, створеним у VS2019, збирати та завантажувати прошивку на плату Arduino Uno.

Також далі буде продемонстровано, як створювати, компілювати та завантажувати програму за допомогою VS2019 та MegunoLink Pro.

3.4 Тестування коректності завантаження прошивки через середовище VS2019

На наступному кроці перевіримо коректність завантаження прошивки через середовище VS2019. З цією метою використаємо описану вище тестову програму Blink. Як було зазначено вище мостом між середовищем VS2019 та Arduino є програма MegunoLink Pro.

Будь-яка програма, створена за допомогою VS2019, повинна бути частиною проекту. Потрібно створити проект, виконавши три кроки:

1. Вибрати Файл → Новий → Проект → Шаблони → Visual C ++ → Програма Arduino.

2. Вказати назву нового проекту. Для цього прикладу було обрано назву "HelloArduino". Потрібно використати розташування за замовчуванням для створення каталогу проекту. Рішення автоматично буде збігатися з назвою проекту.

3. Натиснути ОК.

На рисунку 3.10 показано екран проекту «HelloArduino» з кодом для програми .cpp, яка відображається у головному вікні редактора.

Program.cpp – це ім'я за замовчуванням, яке автоматично надається VS2019 як частина шаблону VC ++ / Arduino.

Він також містить робочий код програми для миготіння світлодіодом, ця програма використовує світлодіод, підключений до виводу 13, як це було для першої програми, що мала назву “Blink”, показаної раніше.

Потрібно вибрати пункт Побудувати → Build Solution для компіляції коду. Можна також просто натиснути функціональну клавішу F7, щоб негайно зібрати проект. Створений код, який створюється вперше, використовує ім'я проекту з розширенням файлу .elf.

Цей тип файлу є виконуваним типом, з стандартним налаштуванням, він автоматично зберігається у папці C: \ Users \ UserName \ My Documents \ Visual Studio2012 \ Projects \ HelloArduino \ HelloArduino \ bin \ debug.

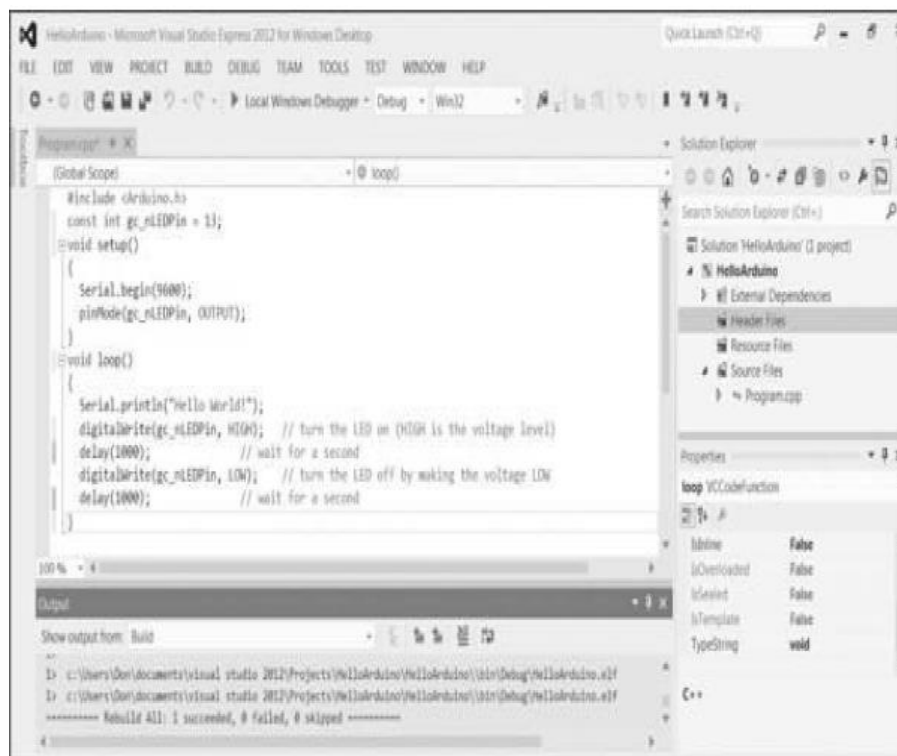


Рисунок 3.10 – Вікно VS2019 з відкритим редактором коду

Програмне забезпечення MegunoLink Pro автоматично створює супутній шістнадцятковий файл, який є єдиним типом, і який можна завантажити в Uno. Новий шістнадцятковий файл також зберігається в тому ж каталозі, що і оригінальний файл з розширенням .elf.

Фактичне завантаження програмою шістнадцяткових файлів здійснюється за допомогою візуалізатора у програмі MegunoLink Pro. Потрібно вибрати візуалізатор пристрою зі списку візуалізаторів та додати в розташування шістнадцятковий файл, після чого вибрати послідовний порт, який було попередньо налаштовано. Також потрібно переконатись, що Arduino Uno з'явився у спадному меню.

На завантаження нещодавно створеного шістнадцяткового файлу в пам'ять Arduino Uno потрібно лише кілька секунд. Графічний індикатор прогресу показує, приблизний час очікування. На рисунку 3.10 показано вікно візуалізатора програмних пристроїв.

Після кожної модифікації програми необхідно перекомпілювати код у VS2019 та завантажити його у пам'ять пристрою за допомогою програмного забезпечення MegunoLink Pro.

3.5 Написання модифікованої прошивки для програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

Додамо до наявної реалізації програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino деякі функції, крім базової функціональності, реалізованої вище. Створимо функції встановлення та підтвердження пароля, які дозволять використовувати гаражні ворота через відкрите з'єднання Wi-Fi без ризику несанкціонованого керування гаражними дверима.

Як зазначалося вище, цей проект використовуватиме VS2019 IDE, а також програмне забезпечення MegunoLink Pro.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						59
Зм.	Арк.	№докум.	Підпис	Дата		

Для реалізації проекту створено файл рішень VS2019 Garage Door Opener.sln, що знаходиться в директорії GarageDoorOpener-master. У цьому ж каталозі створено ще один файл із назвою Garage door opener config interface.mix, який знадобиться при встановленні паролів.

Проект “Garage Door Opener” можна створити у VS2019, виконавши наступні дії:

- 1) вибрати пункт Файл → Відкрити проект;
- 2) перейти до файлу рішення;
- 3) вибрати файл рішення (Garage Door Opener.sln);
- 4) натиснути ОК.

На рисунку 3.11 показано цей проект із створеним файлом програми program.cpp, який відображається у вікні редактора коду.

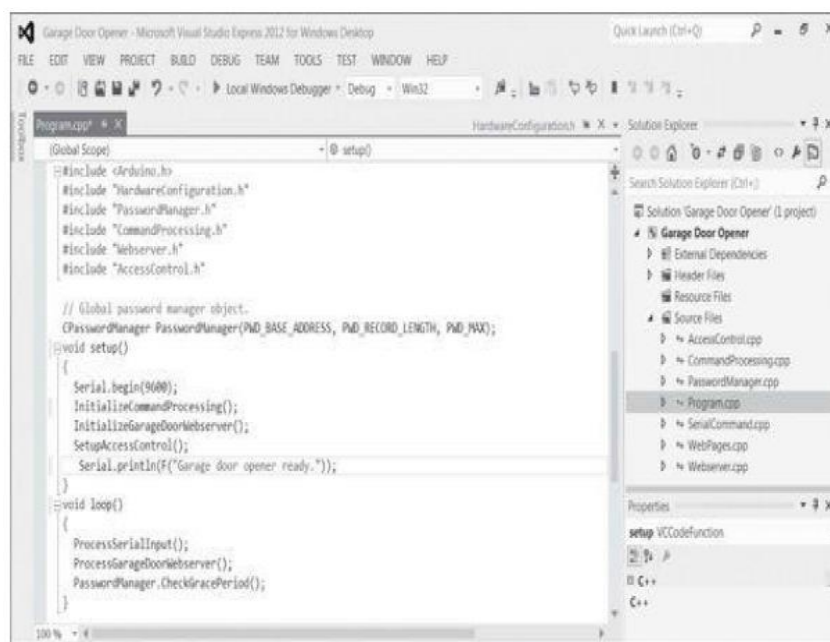


Рисунок 3.11 – VS2019 IDE з запущеним проектом “Garage Door Opener”

Варто зауважити, що Program.cpp містить лише два методи, loop та setup, які також є основою всіх інших ескізів Arduino.

Однак існує значна кількість додаткових файлів, які забезпечують додаткову функціональність, необхідну для підтримки доступу до Інтернету, а

також захисту паролем. Список цих додаткових файлів наведено у провіднику рішень на правій панелі VS2019.

На наступному кроці необхідно внести деякі зміни конфігурації всередині файлу Configuration.h, щоб отримати функціональний проект. Цей файл є заголовковим файлом. Файли цього типу містять додаткову інформацію, яка має вирішальне значення для підтримки файлів проекту, тобто файлів, які закінчуються розширенням .cpp. З цією метою потрібно двічі клацнути на назві файлу, що відображається у списку включених файлів, зображеному на рисунку 3.12.

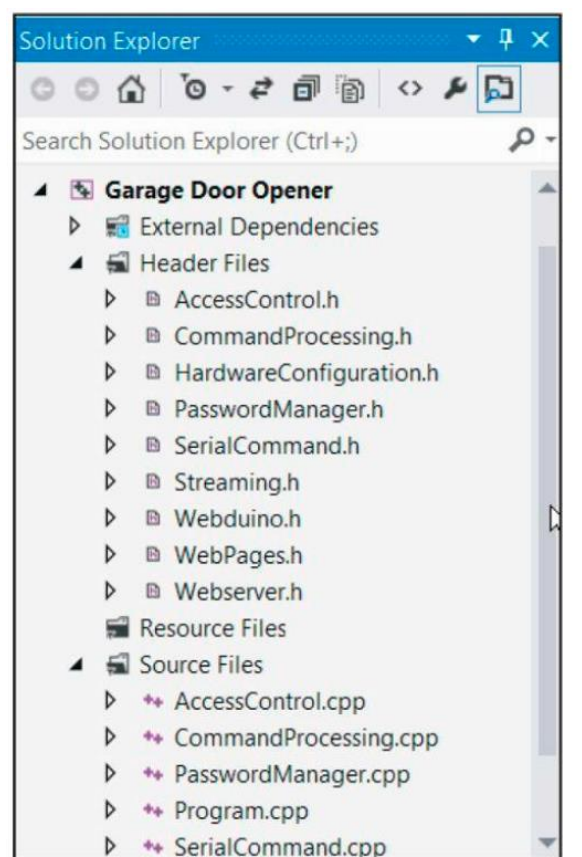


Рисунок 3.12 – Список файлів проекту

Далі необхідно змінити константу LOCAL_IP_ADDRESS [38], щоб відповідати призначеній IP-адресі Arduino Uno.

Також є можливість призначити унікальну MAC-адресу, якщо вона вказана на платі, в іншому випадку можна використовувати адресу за замовчуванням.

Можна також змінити максимально можливу кількість паролів, яка встановлена, на 10, і максимальну довжину пароля, яка встановлена, на 16 символів.

Проте не рекомендовано змінювати жоден із цих параметрів. Паролі встановлюються за допомогою спеціальної програми.

Вони постійно зберігаються в пам'яті EEPROM Uno [39, 40], щоб забезпечити миттєвий доступ під час запуску основної програми програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino. Далі потрібно зберегти змінений файл Configuration.h, перш ніж продовжувати компіляцію.

На наступному кроці потрібно вибрати пункт Побудувати → Build Solution для створення шістнадцяткового файлу, який буде завантажено в Uno. Після чого потрібно використовувати програму візуалізатора пристрою MegunoLink Pro для завантаження шістнадцяткового файлу в Uno.

Наступним кроком є встановлення принаймні одного пароля для тестування цього проекту. Паролі встановлюються за допомогою файлу конфігурації MegunoLink Pro. Все, що потрібно зробити, це двічі клацнути файл Garage door opener config interface.mih, який знаходиться в тому ж каталозі, що і файл рішення проекту. На рисунку 3.13 показано екран MegunoLink Pro, який з'являється під час запуску цього файлу.

Спочатку потрібно вибрати кнопку ініціалізації, яка готує EEPROM Uno до прийому нових паролів. Далі необхідно вибрати один із десяти слотів, в якому буде зберігатися встановлений пароль.

Перший слот повинен бути попередньо вибраний. Крім того, у текстовому полі для введення нового пароля з'являється пароль за замовчуванням. В даному випадку було використано саме цей пароль за замовчуванням, оскільки цього достатньо для продовження початкового тесту.

					КВРКІ.170294.17.02.24 ПЗ	Арк.
						62
Зм.	Арк.	№докум.	Підпис	Дата		

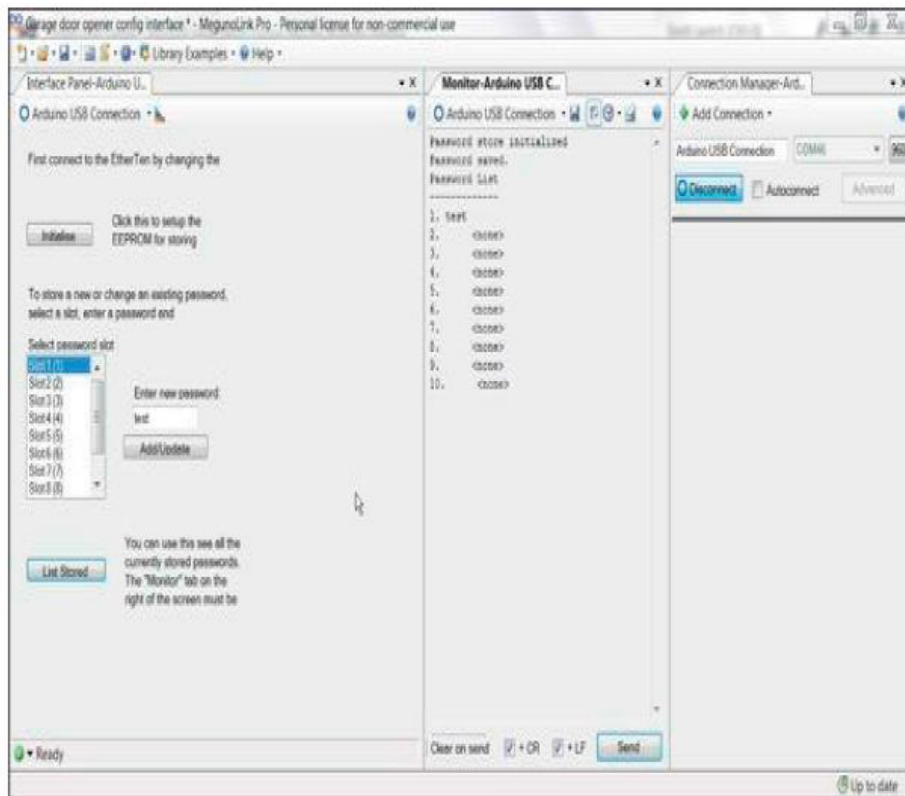


Рисунок 3.13 – Вікно конфігуратора паролів

Далі необхідно натиснути кнопку Додати / Оновити, щоб зберегти пароль у вибраному слоті.

Можна повторювати цей процес, поки не буде досягнуто обмеження в десять паролів. Користувач завжди може перезаписати будь-який пароль, просто призначивши новий пароль для даного слота та вибравши кнопку Додати / Оновити.

3.6 Тестування коректності роботи модифікованої прошивки програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino

Перепрограмований Arduino Uno з ланцюгом керування реле був встановлений в ту саму коробку, в якій зберігалися початкова схема проекту. Для браузера було встановлено IP-адресу Uno, також була відкрита веб-сторінка, інтерфейс якої наведений на рисунку 3.14.

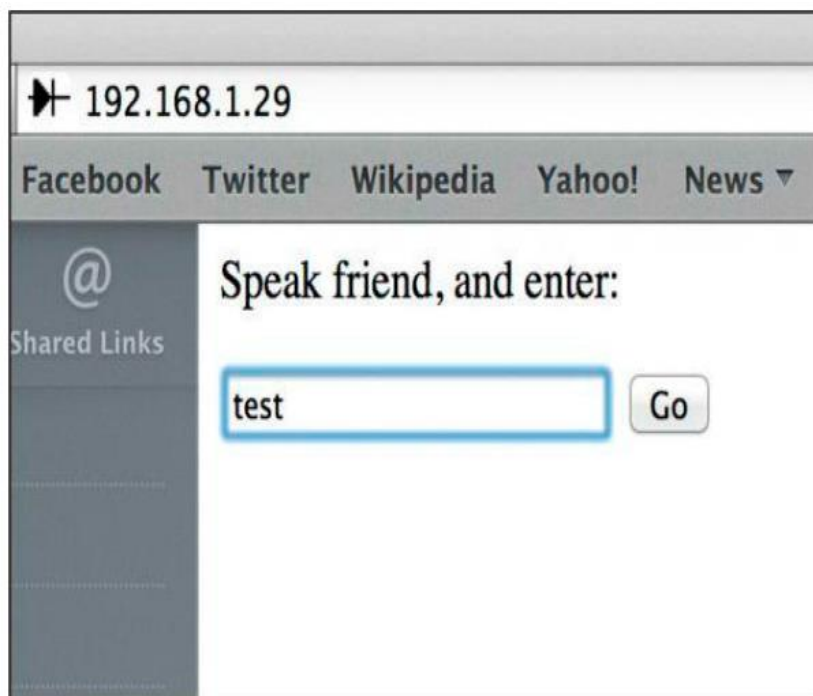


Рисунок 3.14 – Головна сторінка з полем для вводу паролю для керування програмно-апаратним пристроєм віддаленого керування системою відкриття гаражних воріт на базі Arduino

Далі необхідно ввести будь-який із збережених паролів для того, щоб з’явилась веб-сторінка для активації програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino, як видно з рисунку 3.15.

Вибір кнопки «Активувати двері» успішно відкрив ворота, а кнопка “Закрити” їх закрила, як і очікувалося.

Також було протестовано можливість отримати доступ до програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino за допомогою смартфона, як можна бачити на рисунку 3.16.

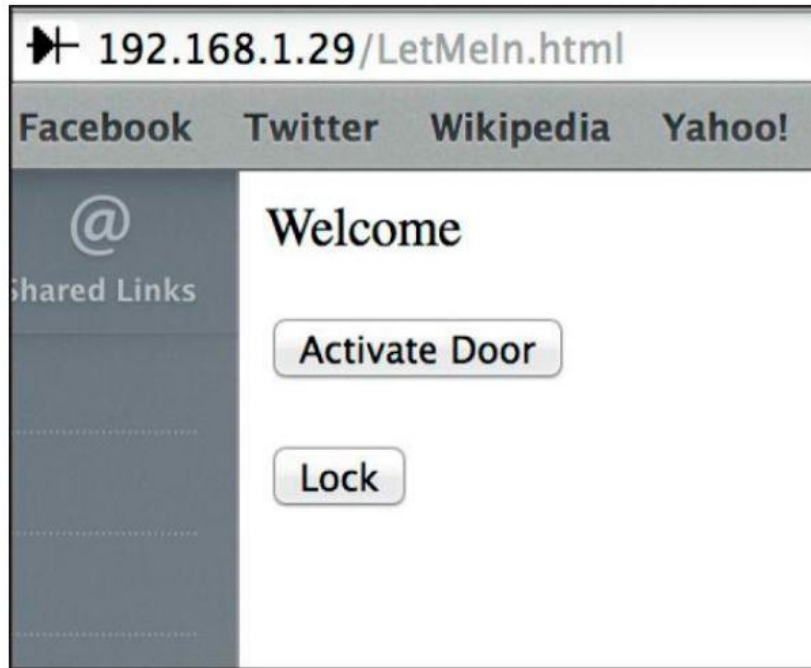


Рисунок 3.15 – Сторінка з кнопками керування програмно-апаратним пристроєм віддаленого керування системою відкриття гаражних воріт на базі Arduino



Рисунок 3.16 – Доступ до сторінки керування програмно-апаратним пристроєм віддаленого керування системою відкриття гаражних воріт на базі Arduino з мобільного пристрою

3.7 Висновки

У третьому розділі було створено прототип програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino. З цією метою було розміщено на макетній дошці реле, зв'язка плат Arduino Uno та Ethernet Shield та поміщено їх в пластиковий корпус.

Далі було розроблено програмне забезпечення для програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino за допомогою середовища розробки Visual Studio 2019 (VS2019) від Microsoft.

В якості моста між середовищем VS2019 та Arduino було використано програмне забезпечення MegunoLink Pro, що надало можливість завантажити скомпільований проект VS2019 на плату Arduino. Поєднання цих двох середовищ розробки є дуже потужним інструментом, оскільки надає можливість отримати повний доступ до середовища розробки C/C++.

За основу програмної реалізації було використано базове проектне рішення, створене в розділі 2, яке було модифіковане для реалізації програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Таким чином, було розроблено та протестовано програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino, що для забезпечення безпеки використовує автентифікацію користувача шляхом застосування паролів.

					КВРКІ.170294.17.02.24 ПЗ	Арк
						66
Зм.	Арк.	№докум.	Підпис	Дата		

ВИСНОВКИ

В повсякденне життя людини активно впроваджуються інноваційні технології, зокрема системи домашньої автоматизації. Системи домашньої автоматизації надають можливість підвищити комфорт та безпеку користувачів шляхом можливості управління роботою різних пристроїв у домі. Доступність використання будинкового обладнання є надзвичайно важливою функцією також для літніх людей та людей з обмеженими можливостями.

Основними недоліками комерційних систем домашньої автоматизації є складність монтажу та завищена вартість таких систем, а також неможливість додавання власних функцій управління і безпеки. Тому актуальною є задача проектування власних пристроїв для домашньої автоматизації, які б не поступалися за функційними характеристиками комерційним системам, але при тому дозволяли б усунути перераховані вище недоліки.

Тому метою роботи є проектування та реалізація програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Об'єктом дослідження є програмно-технічний (апаратний) засіб – програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Предметом дослідження є програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino.

В першому розділі кваліфікаційної роботи було проведено аналіз функцій систем домашньої автоматизації, а також відомих комерційних систем розумної автоматики для гаражних воріт як складової систем домашньої автоматизації. Також було досліджено основні архітектурні особливості апаратної платформи Arduino та особливості інтеграції сервоприводів в системи розумної автоматики для гаражних воріт.

На основі проведеного дослідження було визначено задачі, які необхідно вирішити для проектування власного програмно-апаратного пристрою

					КВРКІ.170294.17.02.24 ПЗ	Арк
						67
Зм.	Арк.	№докум.	Підпис	Дата		

віддаленого керування системою відкриття гаражних воріт на базі Arduino, в якому усунуто перераховані недоліки комерційних систем розумної автоматики для гаражних воріт.

В другому розділі було проведено вибір складових для проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino, а також обґрунтовано цей вибір. Також було обрано програмне забезпечення для програмування плати Arduino Uno rev 3. Після цього було здійснено підключення плат керування Ethernet Shield та Arduino Uno та протестовано коректність їх спільної роботи.

У третьому розділі було створено прототип програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino. З цією метою було розміщено на макетній дошці реле, зв'язку плат Arduino Uno та Ethernet Shield та поміщено їх в пластиковий корпус.

Далі було розроблено програмне забезпечення для програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino за допомогою середовища розробки Visual Studio 2019 (VS2019) від Microsoft. В якості моста між середовищем VS2019 та Arduino було використано програмне забезпечення MegunoLink Pro, що надало можливість завантажити скомпільований проект VS2019 на плату Arduino. Поєднання цих двох середовищ розробки є дуже потужним інструментом, оскільки надає можливість отримати повний доступ до середовища розробки C/C++. За основу програмної реалізації було використано базове проектне рішення, створене в розділі 2, яке було модифіковане для реалізації програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

Таким чином, було розроблено та протестовано програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino, що для забезпечення безпеки використовує автентифікацію користувача шляхом застосування паролів.

Практична цінність роботи полягає в спроектованому та реалізованому програмно-апаратному пристрої віддаленого керування системою відкриття гаражних воріт на базі Arduino, в якому враховано недоліки комерційних систем

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		68

розумної автоматики для гаражних воріт та який може бути застосований як складова частина в системах домашньої автоматизації.

					КВРКІ.170294.17.02.24 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		69

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Lea P. Internet of Things for Architects: Architecting IoT solutions by implementing sensors, communication infrastructure, edge computing, analytics, and security. Birmingham: Packt Publishing, 2018. 524p.
2. Hassan Q. F., Khan A. R., Madani S. A. Internet of Things: Challenges, Advances, and Applications. Florida: CRC Press, 2017. 436 p.
3. Розумний будинок – з чого він складається та чи потрібен вам. URL: <https://nachasi.com/tech/2018/06/25/smart-house-faq/> (дата звернення: 21.05.2021).
4. Bhattacharjee S. Practical Industrial Internet of Things Security. Birmingham, United Kingdom: Packt Publishing Ltd 2018. 324 p.
5. Овчинников Н.А., Мисюрина К.В., Рудникова М.Н., Максимова Е.А. Формализованная модель информационной безопасности системы «Умный дом». Ежемесячный научно-практический журнал – Махачкала: НИЦ «Апробация» 2016. №1. с. 49–51
6. Smart House. Дистанційне керування будинком. URL: <http://hifidom.com.ua/statti/smarthome/distcontrol> (дата звернення: 21.05.2021).
7. Ганжа В. Л. Основы эффективного использования энергоресурсов: теория и практика энергосбережения. Минск: Белорусская наука 2017. 451 с.
8. Тесля Е. А. Умный дом своими руками. Строим интеллектуальную цифровую систему в своей квартире. Санкт Петербург: НТ Пресс, 2018. 224 с.
9. Solanki V., Garcia Diaz V., Davim J. P. Handbook of IoT and Big Data. Florida: CRC Press, 2019. 356 p.
10. Javed A. Building Arduino Projects for the Internet of Things. New York City: Apress, 2016. 285 p.
11. Geddes M. Arduino Project Handbook. San Francisco: No Starch Press, 2016. 272 p.
12. Geddes M. Arduino Project Handbook, Volume 2: 25 Simple Electronics Projects for Beginners. San Francisco: No Starch Press, 2017. 272 p.

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		70

13. Turner R. Arduino Programming: The Ultimate Intermediate Guide to Learn Arduino Programming Step by Step. Amazon Digital Services LLC – KDP Print US, 2019. 337 p.

14. Hughes J. M. Arduino: A Technical Reference: A Handbook for Technicians, Engineers, and Makers. Sebastopol, California: O'Reilly Media 2016. 638 p.

15. Knight S. Arduino for Beginners: Step-By-Step Guide to Arduino (Arduino Hardware & Software). Amazon Digital Services LLC – KDP Print US, 2018. 140 p.

16. Jadot F. Smart Control Harnesses the Industrial Internet of Things. MACHine and Process Management 2016. June 30. 3 p.

17. Arduino Uno R3 Technical Details. URL: http://static6.arrow.com/aopdfconversion/97b354c3cd11851042ebd1fd252fbd6b6def5a73/adafruit2877arduino_uno.pdf (дата звернення: 18.05.2021).

18. Матвієнко М. П. Комп'ютерна схемотехніка: навчальний посібник. Київ: ТОВ "Центр навчальної літератури", 2012. 190 с.

19. Ключев А. С., Ротач В. Я., Кузицин В. Ф. Автоматизация настройки систем управления. Москва: Альянс, 2015. 272 с.

20. Smart Home With Arduino. URL: <https://www.instructables.com/Smart-home-with-arduino/> (дата звернення: 21.05.2021).

21. Arduino programming notebook. URL: <http://engineering.nyu.edu/gk12/amps-cbri/pdf/ArduinoBooks/Arduino%20Programming%20Notebook.pdf> (дата звернення: 18.05.2021).

22. Comunello. Комплект автоматики для гаражних ворот Rampart RT1000LKIT. URL: <https://f.ua/ua/comunello/komplekt-avtomatiki-dlja-garajniyh-vorot-rampart-rt1000lkit.html> (дата звернення 05.04.2021).

23. Умный дом. Послушный дом. URL: <https://www.apple.com/ru/ios/home/> (дата звернення 01.04.2021).

24. Hörmann. URL: <https://www.hormann.ua/> (дата звернення 17.0.2021).

25. Alutech. URL: <https://alutech-group.com/> (дата звернення 10.04.2021).

26. Arduino. URL: <https://www.arduino.cc> (дата звернення: 21.05.2021).

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		71

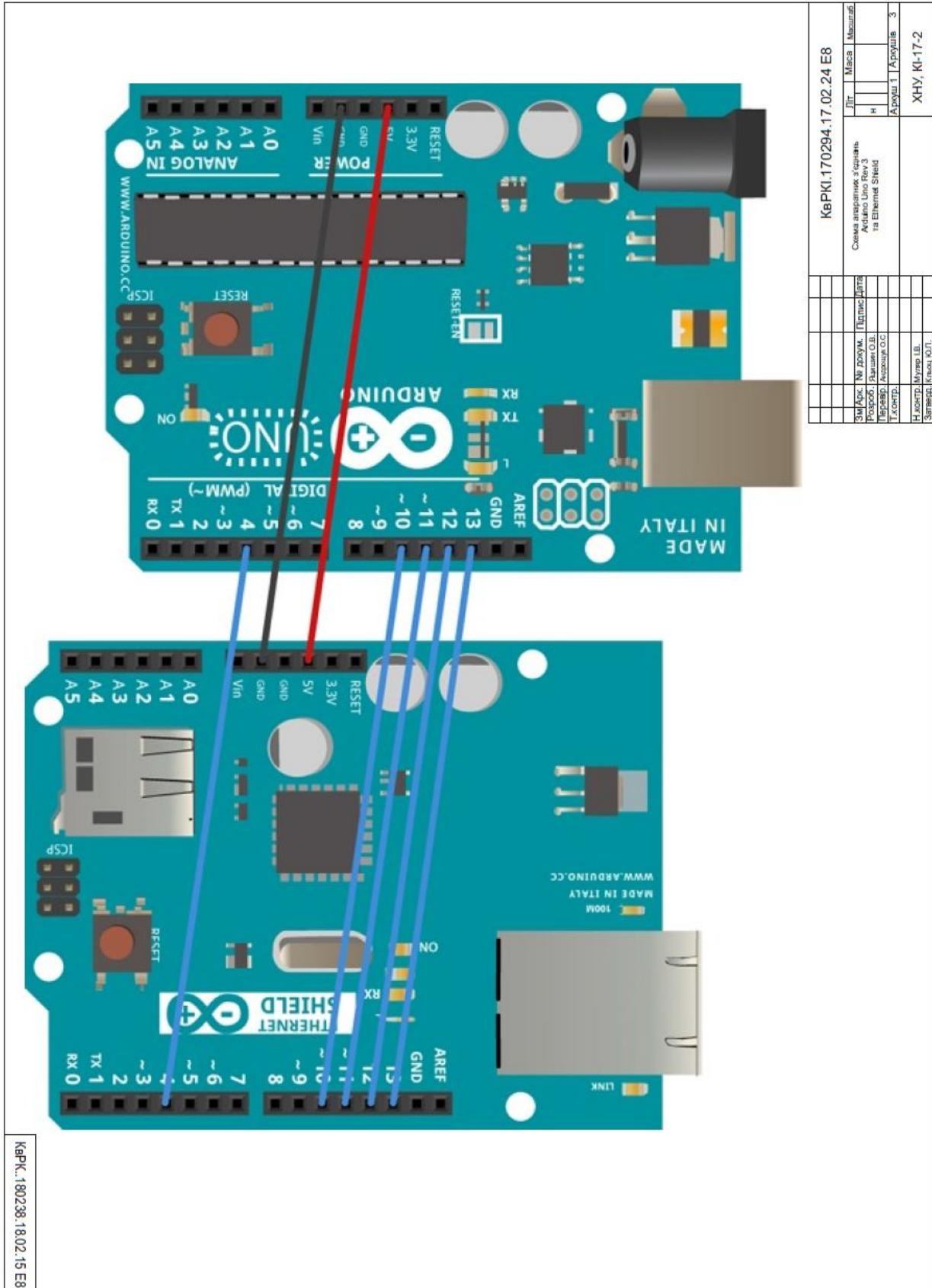
27. Шилд W5100 ethernet shield. URL:<https://arduino.ua/prod391-w5100-ethernet-shield> (дата звернення 21.05.20021).
28. Ethernet Shield и Arduino – основы. URL:<https://arduino-diy.com/arduino-ethernet-shield-osnovi> (дата звернення 22.05.2021).
29. Desktop IDEs. URL:<https://www.eclipse.org/ide/> (дата завернення 23.04.2021).
30. IDE Releases. URL:<https://www.arduino.cc/en/main/OldSoftwareReleases> (дата звернення 12.05.2021).
31. Побудова системи "Розумний будинок" на базі Arduino. URL:<https://card-file.onaft.edu.ua/handle/123456789/10929> (дата звернення: 24.05.2021).
32. Програмування мікроконтролерів. З чого почати. URL : <http://eprints.zu.edu.ua/25722/1/Кузьменко%20С.%20В..pdf> (дата звернення: 24.05.2021).
33. 2N3904, Транзистор: NPN; биполярный. URL:<https://eltis.ua/catalog/tranzistory-npn-tht33ух/2n3904-tranzistor-npn-bipolyarnyy-40v-200ma-625mvt-to92> (дата звернення 22.05.2021).
34. Макетна плата безпачна MB-102 400 отворів. URL:<https://arduino.ua/prod218-maketnaya-plata-bespaechnaya-mb-102-400-tochek> (дата звернення 04.04.2021).
35. Корпус для Arduino Rev 3. URL : http://geekmatic.in.ua/acrylic_box_for_arduino_uno_r3 (дата звернення 05.05.2021).
36. Visual Studio 2019. URL : <https://visualstudio.microsoft.com/ru/vs/> (дата звернення 07.04.2021).
37. MegunoLINK. URL:<https://www.megunolink.com/> (дата звернення 09.04.2021).
38. Crate local_ip_address. URL: https://docs.rs/local-ip-address/0.3.0/local_ip_address/ (дата звернення 07.05.2021).
39. Библиотека EEPROM. URL : <https://doc.arduino.ua/ru/prog/EEPROM> (дата звернення 25.05.2021).
40. Arduino: EEPROM – работа с энергонезависимой памятью. URL : http://codius.ru/articles/Arduino_EEPROM_работа_с_энергонезависимой_памятью (дата звернення 27.05.2021).

					КВРКІ.170294.17.02.24 ПЗ	Арк
Зм.	Арк.	№докум.	Підпис	Дата		72

ДОДАТОК А

(обов'язковий)

Копія креслення «Схема апаратних з'єднань Arduino Uno Rev3 та Ethernet Shield»



ДОДАТОК В

(обов'язковий)

Копія креслення «Інтерфейсні вікна програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino»

КапК.180238.18.02.15.E8

КапК.170294.17.02.24.E8	
Інтерфейс для розробленого пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino	ХНУ, КІ-17-2
СМ.Апр. № докум. Підпис/Дата Розроб. Підпис/ОБ. Перев. Підпис/ОС Тех.впр.	Лист Мес-я Н Аркуш 3 з 3
Н.контр. Микола ІВ. Затверд. Ірина ВОІ.	

75

Додаток Г

Лістинг коду для прошивки плати Arduino

```
#include <SPI.h>
#include <Ethernet.h>
Byte mac[] = { 0xDE, 0xAD, 0xEE, 0xEF, 0xFE, 0xED };
IPAddress ip (10,0,0,20); // IP address, may need to change depending on network
EthernetServer aerver (80); // create a server at port 80
String HTTP_req; // stores the HTTP request
boolean led_status = 0; //state of LED, off by default
Void setup()
{
Ethernet.begin (mac,ip); //initialize Ethernet device
server.begin(); // start t listen for clients
Serial.begin(9600); // for daignostics
pinMode (2, OUTPUT); // LED on pin 2
}
void loop ()
{
EthernetClient client = server.available (); // try to get a client
If (client) { // got a client?
boolean currentLineIsBlank = true;
while ( client.available () ) { // client data available to read
char c = client.read (); // read 1 byte (character) from client
HTTP_req += C; // seve the HTTP request 1 char at a time
// last line of client request is blank and ends with \n
// respond to client only after last line received
if (c == '\n' && currentLineIsBlank){
//send a standard http response header
client.println ("HTTP/1.1 200 OK");
```

```

client.println ("Content-Type: text/html");
client.println ("Connection: close");
client.println ();
//send web page
client.println ("<!DOCTYPE html>");
client.println ("<html>");
client.println ("<head>");
client.println ( "<title>Arduino LED Control </title>");
client.println ("</head>");
client.println ("<body>");
client.println ("<h1>LED</h1>");
client.println ("<p>Select to avitch LED on and off.</p>");
client.println ("<form method='get'>");
ProcessCheckbox(client);
client.println ( "</form>");
client.println ( "</body>");
client.println ("</html>");
Serial.print (HTTP_req);
HTTP_req = ""; // finished with request, empty string
break;
}
//every line or text received from the client ands with \r\n
If (c == '\n') {
// last character on line of received text
// starting new line with next character read
currentLineIsDlank = true;
}
else if (c != '\r'){
// a text character was received from client
currentLineIsBlank = false;
}
}

```

```

} // end if (client.available())
} // end while (client.connected())
delay (1); // give the web browser time to receive the data
client.stop(); // close the connection
} // end if (client)
}
// switch LED and send back HTML for LED checkbox
void ProcessCheckbox(EthernetClient cl)
{
if (HTTP_req.indexOf ("LED2=2") > -1) { // see if checkbox was selected
// the checkbox was selected, toggle the LED
if (LED_status) {
LED_status = 0;
}
else {
LED_status = 1;
}
}
if (LED_status) { //switch LED on
digitalWrite(2, HIGH);
// checkbox is checked
cl.println("<input type =\"checkbox\" name = \"LED2\" value = \"2\" \"
onselect = \"submit();\"checked>LED2");
}
else { // switch LED off
digitalWrite (2,LOW);
// checkbox is unchecked
c1.println("<input type =\"checkbox\" name =\"LED2\" value = \"2\" \"
onselect=\"submit();\"LED2");
}
}
}

```

Лістинг модифікованого коду для прошивки плати Arduino

```
#include <SPI.h>
#include <Ethernet.h>
//MAC address from Ethernet Shield sticker under board
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xED};
IPAddress ip(192, 168, 1, 29); // IP address, may need to change depending on
network
EthernetServer server (80); // create a server at port 80
String HTTP_req; // stored the HTTP request
boolean LED_status = 0; // state of LED, off by default
void setup()
{
Ethernet.begin (mac,ip); // initialize Ethernet device
server.begin(); // start to listen for clients
Serial.begin(9600); // for diagnostics
pinMode(2,OUTPUT); // to transistor on pin 2
}
void loop
{
EthernetClient client = server.available (); // try to get client
if (client) { // got client?
boolean currentLineIsBlank = true;
while (client.connected()) {

if (client.available()) { // client data available to res
char c = client.read (); // read 1 byte (character) from client
HTTP_req += c; // save the HTTP request 1 char at a time
// last line of client request is blank and ends with \n
// respond to client only after last line received
```

```

if (c == '\n' && currentLineIsBlank ){
//send a standard http response header
client.println ( "HTTP/1.1 200 OK");
client.println ( "Content-Type: text/html");
client.println ("Connection: close");
client.println ();
//send web page
client.println ("<!DOCTYPE html>");
client.println ("<html>");
client.println ("<head>");
client.println ("<title>Arduino Garage Door control </title>");
client.println ("</head>");
client.println ("<body>");
client.println ("<h1>Garage Door Opener</h1>");
client.println ("<p>Select to activate door.</p>");
client.println ("<form method = \get\>");
ProcessCheckBox(client);
client.println ("</form>");
client.println ("</body>");
client.println ("</html>");
Serial.print (HTTP_req);
HTTP_req = " "; // finished with request, empty string
break;
}
//every line of text received from the client rnds with \r\n
if (c == '\n'){
//last character on line of received text
// starting new line with next character read
currentLineIsBlank = true;
}
else if ( c != '\r' ) {

```

```

// a text character was received from client
currentLineIsBlank = false;
}
} // end if (client.available())
} // end while (client.connected ())
delay(1); // give the web browser time to receive the data
client.stop(); // close the connection
} // end if (client)
}

// turn on relay and send back HTML for LED checkbox
void ProcessCheckbox(EthernetClient c1)
{
if (HTTP_req.indexOf ("DOOR=2") > -1) { // see if checkbox was selected
// the checkbox was selected, toggle the LED
if (LED_status) {
LED_status = 0;
}
else {
LED_status = 1;
}
}

// keep the logic but it now does the same action no matter when selected
if (LED_status) { // turn relay on for 1 second
digitalWrite(2, HIGH);
delay(1000);
digitalWrite(2, LOW);
// checkbox is unchecked
cl.println ("<input type = \"checkbox\" name = \"DOOR\" value = \"2\" \
onselect = \"submit(); \">>DOOR");
}
}
}

```



User name:
Кафедра кибербезпеки

Check ID:
1008362433

Check date:
28.06.2021 16:12:20 EEST

Check type:
Doc vs Internet

Report date:
28.06.2021 16:12:58 EEST

User ID:
100005590

File name: **Яцишин бакалаврська**

Page count: **69** Word count: **10461** Character count: **81488** File size: **7.62 MB** File ID: **1008431990**

1.36% Matches

Highest match: **0.68%** with Internet source (https://essuir.sumdu.edu.ua/bitstream-download/123456789/71281/1/Shergin_Master)

1.36% Internet sources 26

Page 71

No Library search was conducted

0% Quotes

Exclusion of quotes is off

Exclusion of references is off

0% Exclusions

No exclusions

Modifind

Text modifications detected. Find more details in the online report.

Replaced characters 12

Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 0.0%

Словари проверки: en_US, ru_RU, ua_UA. Ошибка в документах: 11%

ID: 95456 Название: Програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino Добавлено в БД: 2021-06-28 Авторы: О.В. Яцишин Руководители: О.С. Андрощук Консультанты: Оponentы:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	71832	604	186 (0%)	7 (1%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Яцишин Олександр Васильович

Тема: Програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3; кількість сторінок записки 69

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є проектування та реалізація програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino.

2. Висновок про відповідність роботи дипломному завданню: Кваліфікаційна робота повністю відповідає поставленому завданню.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: Розділ 1 – проведено аналіз функцій систем домашньої автоматизації, основних архітектурних особливостей апаратної платформи Arduino, особливостей інтеграції сервоприводів в системи розумної автоматики для гаражних воріт, а також відомих комерційних систем розумної автоматики для гаражних воріт та визначено задачі кваліфікаційної роботи. Розділ 2 – проведено вибір складових для проектування програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino, а також обґрунтовано цей вибір. Здійснено підключення плат Ethernet Shield та Arduino Uno та протестовано коректність їх спільної роботи. Розділ 3 – створено прототип програмно-апаратного пристрою віддаленого керування системою відкриття гаражних воріт на базі Arduino та розроблено відповідне програмне забезпечення.

4. Позитивні сторони роботи: Розроблений програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino може бути застосований як складова частина в системах домашньої автоматизації.

5. Негативні сторони роботи: В рамках кваліфікаційної роботи варто було приділити більшу увагу аналізу відомих комерційних рішень для відкриття гаражних воріт.

6. Оцінка графічного оформлення та пояснювальної записки роботи: Оформлення пояснювальної записки відповідає діючим стандартам оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на задовільному науково-технічному рівні.

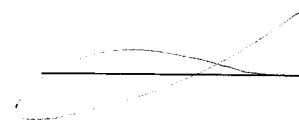
8. Інші зауваження: _____

9. Оцінка дипломної роботи: задовільно.

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи)

Безруков І.В. зав. кафедрою ІІІ, ХНУ

“19” червня 2021 р.

 (підпис)

Яцишина О.В.

ІІБ здобувача вищої освіти

ФПКТС, 4 курсу, групи КІ-17-2

ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність плагіату ознайомлений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів(Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

29.06.2021

дата


підпис

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Програмно-апаратний пристрій віддаленого керування системою відкриття гаражних воріт на базі Arduino

Автор: Яцишин Олександр Васильович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Андрощук Олександр Степанович, д.т.н., проф.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розмішені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розмішені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

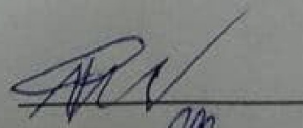
Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення, які мають місце в розділах аналізу існуючих аналогів та прототипів, не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення є фрагментарними, або мають належним чином оформленні посилання;
- 3) в якості запозичень в окремих місцях системою зафіксовано зарезервовані ключові слова мови програмування, які використовуються для розв'язку великої кількості задач і не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 1.36% і адресується до 26 першоджерел, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи



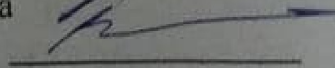
О.С. Андрощук

Гарант ОП



С. М. Лисенко

Завідувач кафедри кібербезпеки та комп'ютерних систем і мереж



Ю.П. Ключ