
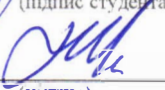



Міністерство освіти і науки України
Український державний університет науки і технологій
Факультет «Управління енергетичними процесами»
Кафедра «Електротехніка та електромеханіка»

Пояснювальна записка

до кваліфікаційної роботи магістра

на тему: «Створення апаратно-програмного комплексу діагностування режимів зарядження тягового акумулятора електромобіля»
зі спеціальності: «141 Електроенергетика, електротехніка та електромеханіка» Виконав: студент групи «ЕП2121»

	 (підпис студента)	/Олексій ВАСИЛЕНКО/ (Ім'я ПРІЗВИЩЕ)
Керівник:	 (підпис)	/зав. кафедри Андрій МУХА/ (посада, Ім'я ПРІЗВИЩЕ)
Нормоконтролер:	 (підпис)	/доц. Оксана КАРЗОВА/ (посада, Ім'я ПРІЗВИЩЕ)
Консультанти:		
_____	 (підпис)	//проф. Олег САБЛІН (посада, Ім'я ПРІЗВИЩЕ)
(назва розділу)		
_____	(підпис)	//
(назва розділу)		
_____	(підпис)	//
(назва розділу)		
_____	(підпис)	//
(назва розділу)		

Засвідчую, що у цій роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент


(підпис)

Дніпро – 2022 рік

Ministry of Education and Science of Ukraine
Ukrainian State University of Science and Technologies

Faculty «Energy And Economic Process Control»
Department «Electrical engineering and electromechanics»

Explanatory Note
to Master's Thesis

on the topic: «Creation of a hardware and software complex for diagnosing the modes of charging of the traction battery of an electric vehicle»
according to educational curriculum «Electromechanical automation systems and electric drive»

in the Speciality: «141 Electrical energetics, electrical engineering and electromechanics»

Done by the student of the group EP2121: /Aleksey VASILENKO/

Scientific Supervisor:

/Andey MUKHA/

Normative controller:

/Oksana KARZOVA/

Supervisors

LP and safety in ES

(Chapter title heading)

/Oleh SABLIN/

(position, name, surname)

//

(Chapter title heading)

(position, name, surname)

//

(Chapter title heading)

(position, name, surname)

//

(Chapter title heading)

(position, name, surname)

Міністерство освіти і науки України
Український державний університет науки і технологій

Факультет: Факультет «Управління енергетичними процесами»


Кафедра: «Електротехніка та електромеханіка»

Рівень вищої освіти: магістр

Освітня програма: «Електромеханічні системи автоматизації та електропривод»

Спеціальність: «141 Електроенергетика, електротехніка та електромеханіка»

ЗАТВЕРДЖУЮ

Завідувач кафедри ЕТЕМ
 /Андрій МУХА/
(підпис)

Дата 01.09.2022 р.

ЗАВДАННЯ

на кваліфікаційну роботу магістра

студенту Василенко Олексій Олександрович

1. Тема роботи: «Створення апаратно-програмного комплексу діагностування режимів зарядження тягового акумулятора електромобіля»

Керівник роботи: Муха Андрій Миколайович,
затверджені наказом № 729 ст від 29.10.2021 р.

2. Строк подання студентом роботи: 01.12.2022 р.

3. Вихідні дані до роботи: Датчики струму з гальванічним розв'язанням
Датчики напруги з гальванічним розв'язанням, система контролю сигналу
«пілот» від електромобілю, система контролю силового ключа зарядного
комплексу, система контролю температури елементів зарядного комплексу,
система контролю параметрів електробезпеки користувачів.


4. Зміст пояснювальної записки (перелік питань, які потрібно опрацювати):

4.1 Аналітична частина: програмування одноплатної комп'ютерної системи,
контролера типу Arduino, створення каналу передачі інформації для
керування програмою на смартфоні або планшеті на ОС Android.

4.2 Основна частина: створення системи для відображення режимів роботи
комплексу за допомогою дисплею, а також виводу інформації на систему
Android для керування системою зарядження.

4.3 Охорона праці та захист навколишнього середовища: вимоги безпеки
під час розробки програмного комплексу зарядження

5. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Завдання видав: (підпис консультанта, дата)	Завдання прийняв: (підпис студента, дата)
Основна частина	зав. кафедри Муха А.М.	 01.09.22	 01.09.22
Охорона праці та безпека при НС	проф. Саблін О.І.	 28.10.22	 28.10.22

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вступ, основна частина – 30%	01.09.22	
2	Основна частина – 60%	14.11.22	
3	Охорона праці та безпека в надзвичайних ситуаціях	27.11.22	
4	Графічні роботи, 100% готовності диплому	30.11.22	
5	Подання кваліфікаційної роботи до кафедри	01.12.22	
6	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	21.12.22	

Студент


(підпис)

Олексій ВАСИЛЕНКО

(Ім'я ПРІЗВИЩЕ)

Керівник роботи


(підпис)

зав. кафедри Андрій МУХА

(Ім'я ПРІЗВИЩЕ)

№	Формат	Позначення	Назва	Кільк.	№ екз.	Прим
1			<u>Документація загальна</u>			
2			Заново розроблена			
3	A4	7.141.170033.ПЗ	Пояснювальна записка	63		
4						
5			Запозичена			
6						
7			<u>Графічна частина</u>			
8			Заново розроблена			
9	A4	7.141.170033.01	Класифікація апаратно-	1		
10			програмних комплексів			
11	A4	7.141.170033.02	Переваги серії	1		
12			мікроконтролерів ESP-32			
13	A4	7.141.170033.04	Код інтерфейсу програми			
14	A4	7.141.170033.03	Інтерфейс розробленої	1		
15			програми у смартфоні	1		
16						
17						
18						
19						
20						
21			Запозичена			
22						
23			<u>Електронна частина</u>			
24						

7.141.170033.ВР							
Змн.	Лист	№ Докум.	Підпис	Дата			
Розроб.		Василенко О.О.	<i>А.Василенко</i>	01.12.22	Літ.	Лист	Листів
Перевір.		Муха А.М.	<i>А.Муха</i>	12.12.22		5	81
Н. Контр.		Карзова О.О.	<i>О.Карзова</i>	14.12.22	Відомість кваліфікаційної роботи		
Затв.		Муха А.М.	<i>А.Муха</i>	14.12.22			
					МОН України, УДУНТ Кафедра ЕТЕМ, група ЕП2121		

Реферат

Дипломна кваліфікаційна робота на тему «Створення апаратно-програмного комплексу діагностування режимів зарядження тягового акумулятора електромобіля» містить: 63 сторінки основного тексту, 61 рисунок, 31 літературне джерело.

Метою роботи є розробка системи зарядження електромобіля від електричної мережі промислової частоти 50 Гц, напругою 220 В з можливістю контролю параметрів зарядження та передачі їх на зовнішній пристрій, який працює з використанням ОС Android.

У **розділі 1** розповідається про переваги та недоліки електромобіля, а також про існуючі зарядки для них в даний момент.

Розділ 2 – класифікація мікроконтролерів, їх види.

Розділ 3 доповідає, чому для створення апаратно-програмного комплексу було використано саме мікроконтролер ESP32 Wemos D1 Mini. Також представлено його переваги, недоліки та характеристику.

Розділ 4 представляє інтерфейс програми. Порівнює його з іншими варіантами середовищ розробки та доводить, що використана програма є кращим вибором серед інших.

У **розділі 5** викладається процес розробки програмного коду мікрокомп'ютера. Представлені основні моменти, розповідається про те, що саме цей код створює. Показані малюнки у вигляді скріншотів цих моментів.





В **розділі 6** розглянуто охорону праці та безпеку в надзвичайних ситуаціях

КЛЮЧОВІ СЛОВА: ЕЛЕКТРОМОБІЛЬ, КОМПЛЕКС ЗАРЯДЖЕННЯ, МІКРОКОНТРОЛЕР, МІКРОКОМП'ЮТЕР, ESP-32, АРДУІНО, ЗАРЯДНИЙ ПРИСТРІЙ EVSE, ПЛАТА, ІНТЕГРАЛЬНА СХЕМА, СІМЕЙСТВО AVR, ШІМ-СИГНАЛ

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		6

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 КЛАСИФІКАЦІЯ ЗАРЯДЖУВАЛЬНИХ КОМПЛЕКСІВ ЕЛЕКТРОАВТОМОБІЛЯ.....	10
1.1 Переваги та недоліки електроавтомобіля.....	10
1.2 Типи систем зарядження.....	11
1.3 Американська класифікація зарядок.....	12
1.4 Європейська класифікація зарядок.....	15
1.5 Порівняння класифікацій.....	16
РОЗДІЛ 2 ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО МІКРОКОМП'ЮТЕРИ.....	17
2.1 Про обчислювально-апаратні платформи.....	17
2.2 Класифікація плати Arduino	19
2.3. Основні моделі мікрокомп'ютерів.....	19
РОЗДІЛ 3 СЕРІЯ ESP32 ТА ВИБІР МІКРОКОНТРОЛЛЕРА.....	22
3.1 Серія мікроконтролерів ESP32.....	22
3.2 Переваги ESP32.....	23
3.3 Підстава до вибору мікроконтролера. Характеристика та огляд вибраної моделі.....	24
3.4 Можливості ESP-32.....	26
РОЗДІЛ 4 РОЗРОБКА ІНТЕРФЕЙСУ ПРОГРАМИ.....	27
4.1 Аналоги до Arduino IDE	27
4.2 Вибір програми для розробки.....	28
4.3 Базовий інтерфейс програми Arduino IDE	30

					7.141.170033.ПЗ			
Змн.	Лист.	№ Докум.	Підпис	Дата	Пояснювальна записка	Лім.	Лист	Листів
Розроб.		Василенко О.О.		01.12.22				
Перевір.		Муха А.М.		01.12.22			7	81
Н. Контр.		Карзова О.О.		01.12.22		МОН України, УДУНТ Кафедра ЕТЕМ, група ЕП2121		
Затв.		Муха А.М.		01.12.22				

РОЗДІЛ 5 ПРОГРАМНИЙ КОД.....	31
5.1 Вихідні дані, налаштування початкового стану.....	31
5.2 Розробка інтерфейсу програми.....	35
5.3 Програмування проекту під час виконання змін інтерфейсу.....	37
РОЗДІЛ 6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ.....	57
6.1 Вимоги безпеки праці під час виконання робіт на робочому місці...	57
6.2 Дії працівників в надзвичайних ситуаціях.....	68
ВИСНОВКИ І ЗАГАЛЬНІ РЕКОМЕНДАЦІЇ.....	72
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ.....	73
СПИСОК ГРАФІЧНИХ РОБІТ.....	76

ВСТУП

За минулі декілька років та через події цього року по всьому світу набувають стрімкого розвитку електроавтомобілі. Але це не тільки економічна причина придбання такого автомобіля. Перш за все, багато людей сьогодні задумалися про екологію нашого світу, а електромобіль забезпечує безпеку навколишньому середовищу, адже такий вид транспорту не має шкідливих вихлопів, які є одним з основних джерел забруднення навколишнього середовища.

Мета роботи - розробити систему зарядження електромобіля від електричної мережі промислової частоти 50 Гц, напругою 220 В з можливістю контролю параметрів зарядження та передачі їх на зовнішній пристрій який працює з використання ОС Андроїд. Таким чином буде створено апаратно-програмний комплекс діагностування режимів зарядження тягового акумулятора електромобіля.

					7.141.170033.ПЗ	Арк.
						9
Изм.	Лист	№ докум.	Підпис	Дата		

РОЗДІЛ 1

КЛАСИФІКАЦІЯ ЗАРЯДЖУВАЛЬНИХ КОМПЛЕКСІВ ЕЛЕКТРОАВТОМОБІЛЯ

1.1 Переваги та недоліки електромобіля

Електромобілі мають інші переваги, серед яких:

- низький рівень шуму, суттєво менший, ніж у звичайних автомобілів;
- надійність двигуна навіть при довготривалій експлуатації, а також його довговічність;
- Високий КПД самого електродвигуна (приблизно 95%);

Серед недоліків:

- Висока вартість тягових АКБ;
- велика маса транспортного засобу;

Але, звісно, хотілося б більше розповісти про економічний фактор, адже електромобіль використовує відносно дешевий енергоресурс – електроенергію. Не враховуючи її вартість, те що це одна з головних ресурсів і нашого життя, але і доступ до такої енергії не вимагає серйозних або складних технологій для її зберігання.

Окрім вищесказаних недоліків, не надто впливових на загальну думку про електромобілі, є складний фактор, який стає завадою для ще ширшого розповсюдження електромобілів – обмежений пробіг, оскільки у неї акумуляторів щодо невелика ємність.

На неї впливає і умови навколишнього середовища (температура, наприклад). Акумулятори потребують постійного обслуговування та періодичної заміни. Термін служби акумуляторів зараз оцінюється на рівні п'яти-сім років, після чого вони вимагають заміни. Але за останні десять років ємність акумуляторів була розвинена, збільшена більше, ніж у 5 разів, а їх вартість тільки знизилася удвічі.

					7.141.170033.ПЗ	Арк.
						10
Изм.	Лист	№ докум.	Підпис	Дата		

Отже, цей недолік різко знижує свою увагу. Ресурс тягового акумулятора залежить в першу чергу від ефективного режиму його зарядження, тому і постає проблема створення систем зарядження які мають можливість контролювати та керувати цим процесом, враховуючі можливості існуючої енергосистеми країни.

1.2 Типи систем зарядження

Насамперед, хотілося б розповісти про типи зарядок електромобіля, які вже існують. Не можу не нагадати, що тягова батарея у таких машин використовує постійний струм. Але в нашій електричній мережі використовується змінний струм. Тобто, якщо ви використовуєте зарядну станцію зі змінним струмом, використовується бортовий зарядний пристрій електромобіля, що перетворює змінний струм від зарядної станції в постійному, внаслідок чого заряджається тягова батарея. Але, безпосередньо, швидкість зарядки електромобіля буде визначатися потужністю бортового зарядного пристрою, параметр якого вказаний виробником машини.

Важливо пояснити на прикладі. Уявимо, що ми використовуємо станцію з непоганою потужністю 18 кВт, але бортовий зарядний пристрій нашої машини 2.9 кВт. У такому разі тягова батарея матиме потужність лише 2.9 кВт. Час заряджання можна також визначити самостійно. Для цього достатньо розділити корисну ємність тягової батареї автомобіля на вказану потужність бортового зарядного пристрою.

Щоправда, у реальному житті потрібно буде трохи більше часу, оскільки процес заряджання літій-іонних акумуляторів не лінійний і в процесі зарядки виникають невеликі втрати. Крім того, самим процесом перетворення одного струму в інший здійснюється станцією. До речі, ці станції (рис.1) зазвичай називають зарядними станціями стандартної зарядки електромобіля [1]. Це стосується станцій змінного струму, а постійного – комплексами експрес-зарядки або швидкими зарядними станціями.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		11



Рис.1 – Зарядна станція стандартної зарядки електромобіля

1.3 Американська класифікація зарядок

Хотів також згадати рівні заряджання [2]. За рівнями потужностей є 3 рівні. В Америці вони називаються Level 1, 2, 3.

Перший рівень займають досить повільні зарядки, змінний струм яких не вищий за 16 А, тобто максимальна потужність, на яку можна розраховувати, це 1.92 кВт. Для середнього автомобіля це означає, що очікування до повної зарядки займе приблизно 12 годин. Так, будь-який автомобіль можна заряджати лише встроївши адаптер в розетку.



Рис. 2 – зарядка першого рівня змінного струму

«Під капотом» подібного зарядного пристрою (Рис.2) знаходяться засоби захисту та регулювання струму, які замикають саму ланцюг у момент, коли роз'єм встромлять у гніздо зарядки електромобіля.

Нині існують такі зарядні пристрої не вище 3.3 кВт. Щодо ціни в домашньому виконанні, така зарядка не коштуватиме вище 500\$, на вулиці такі вже не встановлюють, хіба що невелика кількість старих зарядок рівня 1. Детальніше про такі зарядки ми поговоримо нижче, наше завдання полягає в розробці такої зарядки.

Другий рівень це зарядки потужнішого типу, з потужністю до 7кВт (240В, 30А). Відразу хотів би звернути увагу на цінник таких пристроїв, адже тут ціна починається з 500 \$, якщо створювати її в домашній умові, а вуличних від 1 000 \$.

Обидва рівнів зарядок, і перший, і другий, використовують змінний струм, а в електроавтомобілі, як говорилося раніше, є спеціальний зарядний пристрій, який перетворює струм і заряджає акумулятор.

Нижче наводжу приклади, де мені вдалося знайти і зарядний пристрій у вуличному виконанні (Рис. 3.1), і домашньому (Рис. 3.2).



Рис. 3.1 Зарядка другого рівня змінного струму у вуличному виконанні



Рис. 3.2 Зарядка другого рівня змінного струму у домашньому виконанні

Щодо часу зарядки від таких пристроїв тут справи набагато кращі, достатньо всього близько 4 годин для повної зарядки, якщо електромобіль може прийняти близько 6-7 кВт. Однак варто зауважити, що не всі машини мають подібну потужність і для цього може знадобитися бортовий зарядний пристрій набагато дорожчий.

Також є одне питання, для котрого можна привести невеликий приклад. В середньому водій за рік їздить приблизно 20 000 км/год, це 55 км за день.

Для такої автівки, як Nissan Leaf 2013, витрата складатиме приблизно 18 кВтг/100 км, виходить, що один власник використовуватиме 10 кВтг на день. За стандартом бортовий зарядний пристрій має 3.6 кВт. Тобто, простими словами, знадобиться 3 години щодня для заряджання. Цим хотів наголосити, що існує проблема в інфраструктурі вуличних зарядок, точніше, черги на зарядку у громадських місцях. Однак тут хотів би продовжити і розповісти про швидший варіант.

Останнім рівнем зарядок є **третій**, найпотужніший і найшвидший (Рис.4). Тут вже маємо 300-600В, 100А та більше.

Особливістю таких зарядок є те, що за часом зарядити машину до 80% займає трохи більше 30 хвилин. Але після цього для безпеки акумулятора подальша зарядка або взагалі неможлива, або з меншою потужністю[3].

					7.141.170033.BP	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		14



Рис. 4 – Зарядки типу SAE Combo 2



Рис. 5 – Зарядка типу Mennekes

1.4 Європейська класифікація зарядок

У Європі дещо інша класифікація, стандарт ІЕС 6216 має такі потужності:

- Mode 1 - потужність 240В, 16 А;
- Mode 2 – 240В та 32А;
- Mode 3 – 690В, 3-ф. швидка зарядка зі змінним струмом, 64А, у питанні кіловат – 43, проте найчастіше встановлюються зарядки в половину менше, 22 кВт;
- Mode 4 – найшвидша зарядка з постійним струмом, до 600 В та 400А, тобто 240 кВт, не більше.

1.5 Порівняння європейської класифікації зарядок із американською

Таким чином, у Європі схожі стандарти зарядок за потужністю, як і в США, але є відмінність у плані Mode 3 – швидка зарядка зі змінним струмом, а перші 2 типи – зарядки менш швидкі, швидкість яких вимірюється годинами, а потужні здатні зарядити акумулятор якого або електроавтомобіля не більше, ніж за годину.

За підсумками цього розділу, хотів розповісти, що є безліч варіантів для заряджання електромобіля, як і дешеві, але які займають більше часу для зарядки, так і навпаки, можна заряджати електромашину і за пів години, але це буде коштувати значно більше.

					7.141.170033.BP	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		16

РОЗДІЛ 2

ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО МІКРОКОМП'ЮТЕРИ

2.1 Про обчислювально-апаратні платформи

Метою цієї дипломної роботи насамперед розробити систему зарядження електромобіля від електричної мережі. В цьому може допомогти зарядний пристрій EVSE для електромобіля з ESP32 - однопалатний мікрокомп'ютер типу Ардуїно.

Перш за все, варто розказати про те, що таке мікроконтролер, адже без нього неможлива робота жодної побутової чи спеціальної техніки. Це спеціальний прилад, який можна запрограмувати, а його призначенням є використання в будь-яких керуючих пристроях чи інших системах, наприклад, для технологічних процесів.

Взагалі, ардуїно – обчислювально-апаратна платформа [4] для конструювання, з якою можна створити чимало різноманітних електронних приладів своїми руками. Це плата, у якій є центральний мікроконтролер, а також допоміжні компоненти.

Те, як працює ця платформа, можна представити у вигляді декількох етапів :

1. Спершу Arduino приймає дані від сенсорів, наприклад, зчитує інформацію про освітлення, стан повітря тощо.
2. Потім вона проаналізовує ввідні дані.
3. Приймається рішення, тобто передається команда на виконуючі пристрої, після чого виводиться потрібна інформація на дисплей, виконуються потрібні команди.

Тобто, Arduino – мозок нашого пристрою, а для основної роботи, тобто діагностування режимів зарядження електромобіля, буде використано потрібні для цього датчики і інші виконуючі пристрої. Також для роботи є спеціальні елементи – відведені контакти ввід/вивід.

					7.141.170033.BP	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		17

Лишиться вибрати середовище для розробки.

Мікромп'ютер має багато можливостей, наприклад, його застосовують для створення різних інтерактивних об'єктів або для підключення до програмного забезпечення. Плата Arduino має мікроконтролер Atmel AVR та інші елементи, необхідні для обв'язки, для програмування і інтеграції з іншими пристроями.

Багато плат має лінійний стабілізатор напруги (5В або 3.3В). Тактування відбувається на частоті 8 чи 16 МГц з допомогою кварцового резонатора. Їм не потрібен зовнішній програматор, адже в платах вже є завантажувач (bootloader). Він дозволяє прошивати платформу за допомогою USB, що дешевше ніж програматор.



Рис. 6 - процесорна плата Arduino з платами маркетування [5]

Також плати Arduino можуть використовуватися для багатої кількості виводів мікроконтролера як вхідні або вихідні контакти у зовнішніх схемах.

2.2 Класифікація плати Arduino

Можемо взяти за приклад плату Decimila, тут доступно 14 цифрових входів/виходів, а 6 з них можуть згенерувати ШІМ сигнал, а також 6 аналогових входів. Взагалі, ці сигнали доступні на самій платі через штирові роз'єми або контактні майданчики. Існує велика кількість зовнішніх плат розширення, по іншому їх назва – щити. Вони приєднуються до плати Arduino з допомогою штирових роз'ємів.

Плати Arduino були створені так, щоб в взаємодії з ними їх можна було б розширювати за допомогою компонентів (щитів). З штировими роз'ємами, описаними вище, щити підключаються до Arduino

Є також плати зі зменшеним чи спеціальним фактором. Наприклад, це плати Lilypad або Nano. Але величезна кількість різних датчиків та інших пристроїв, сумісних з платою Arduino, створюються сторонніми виробниками. Також вони випускають різні набори з електромеханічними елементами, які створені для спільної роботи з платами.

Протягом еволюції плат Arduino змінювався також і їх дизайн. Спочатку в початкових платах були великі мікроконтролери, які мали чималі габарити. Але згодом їх зменшили, при цьому не змінив форму та розмір. Це дозволило спростити розробку модулів будь-якого Arduino та полегшити підключення цих модулів до плат Arduino.

2.3 Основні моделі мікрокомп'ютерів

Перед тим, як розповісти, який саме мікроконтролер було обрано, хотів би звернути увагу на інші моделі Arduino, які заслуговують не меншої уваги (також додано посилання на документацію).

Першим прикладом хотів би привести Arduino Diecimila [6].

					7.141.170033.BP	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		19

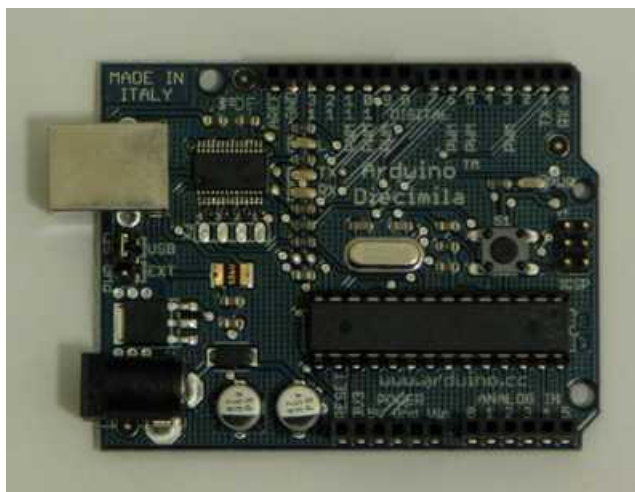
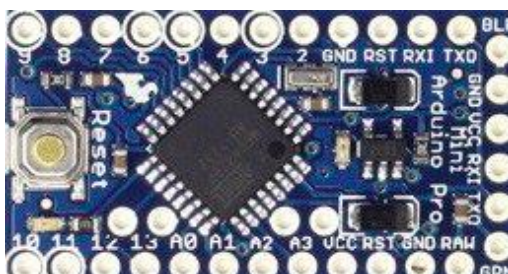


Рис. 7 – процесорна плата типу Arduino Diecimila

Його ми приводили до прикладу раніше. Це плата на основі ATmega168, яка має 14 цифрових входів/виходів, кварцевий генератор в 16 МГц, а також 6 ШІМ-виходів, та 6 аналогових входів. Він містить все, що потрібно для підтримки мікрокомп'ютера, його слід просто підключити до персонального комп'ютера за допомогою USB кабелю, щоб почати програмувати. До речі, «Decimilia» у перекладі з італійської – 10 000. Її так було названо через тираж плат понад 10 000, це остання у серії USB-лат Arduino.

Посилання на документацію – [6]

Також хотів би розповісти про Arduino Pro Mini, які базуються на ATmega328P. Така плата призначена для напівпостійної установки на об'єктах чи виставках.



Ця плата постачається без попередньо встановлених роз'ємів, це дозволяє використовувати інші типи на свій розсуд. Розташування виводів спільне із схожою версією, Arduino Mini.

Документація за посиланням [7]

Також є багато інших моделей, серед яких:

- ArduinoFio - [8];
- Arduino Mega ADK Rev3 – [9];
- Arduino Lilypad Main Board – [10];

					7.141.170033.BP	Арк.
						21
Изм.	Лист	№ докум.	Підпис	Дата		

РОЗДІЛ 3

СЕРІЯ ESP32 ТА ВИБІР МІКРОКОНТРОЛЛЕРА

3.1 Серія мікроконтролерів ESP-32

Серія ESP-32 – низка недорогих за вартістю та з низьким енергоспоживанням мікроконтролерів. Це єдина система на кристалі і влаштованим Wifi і Bluetooth антенами та контролерами.

Ціни на ці пристрої продовжують знижуватися. Однокристальні чіпи знаходять широке застосування у найрізноманітніших сферах: від вимірювальних приладів до виробів розваг та різноманітної домашньої техніки.

На відміну від персональних комп'ютерів мікроконтролер поєднує в одному кристалі функції процесора і периферійних пристроїв, містить оперативну пам'ять і постійний запам'ятовуючий пристрій для зберігання коду і даних, проте має значно менші обчислювальні ресурси. ESP-32 – це мікроконтролер, розроблений компанією Espressif Systems [11]. Це китайський виробник, розташуванням якої є Шанхай. Мікроконтролер є системою на кристалі з інтегрованим Wi-Fi і Bluetooth контролерами.

У серії ESP-32 використовується ядро Tensilica Xtensa LX6. Плати з ESP32 мають гарну обчислювальну здатність, розвинену периферію і при цьому дуже популярні через низьку ціну.

У систему також був влаштований радіочастотний такт: симетричний трансформатор, модулі управління питанням, радіочастотні компоненти, малошумний підсилювач.

					7.141.170033.BP	Арк.
						22
Изм.	Лист	№ докум.	Підпис	Дата		

3.2 Переваги ESP-32

Серія ESP-32 має велику перевагу від аналогічних серій мікроконтролерів через низьку ціну і енергоспоживання, як розповідалось раніше. Крім цього, на базі цієї серії були засновані модульні SMT плати. Ця особливість представляє собою створення прототипів, макетів інших пристроїв на основі цього мікрокомп'ютера, а також без потреби створення самої плати.

Така особливість значно покращує якість створення програмних комплексів та їх підготовчий стан. Ця перевага дозволила серії ESP-32 стати лідером на ринку та зробила її дуже популярною серед аналогів.

Крім того, це створило можливість використовувати мікроконтролери цієї серії для навчання.

Модулі з SMT платами містять ESP-32 SoC. Це полегшує інтеграцію в інші плати. Вимірювані та інтегровані F-антенні конструкції використовують, щоб зробити трасовку антенних друкованих плат на модулях, приклад якої буде представлено нижче. Також певні модулі включають в себе пам'ять, яка зветься псевдостатичною, оперативною (pSRAM).

Прикладом, з яким можна порівняти вибір плати, використаної в цій роботі, можна привести ESP-32 Espressif ESP-WROOM-32. Ця плата на основі мікроконтролера ESP-32 також має світлодіоди, 30 портів, живлення яких 3.3 В, заземлення, а також аналогові та цифрові порти.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		23

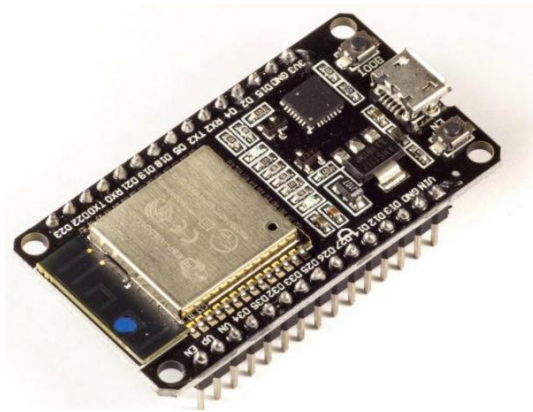


Рис. 8 - модель мікроконтролера Espressif ESP-WROOM-32

3.3 Підстава до вибору мікроконтролера. Характеристика та огляд вибраної моделі

Було обрано саме мікрокомп'ютер ESP-32 Wemos D1 Mini. Справа в тому, що в ньому є вбудований модуль Wifi, котрий дозволяє створити клієнта Wifi мережі або точку доступу Wifi, створювати на собі Web-сервер, приймати та обробляти запити.

Ця плата має в собі сам мікроконтролер ESP-32, Micro-USB, кнопку для перезапуску мікроконтролера та 32 порти.

Важливо також те, що інтегрованість певних елементів управління дозволяє не встановлювати додаткові, що забезпечує цій моделі популярність.

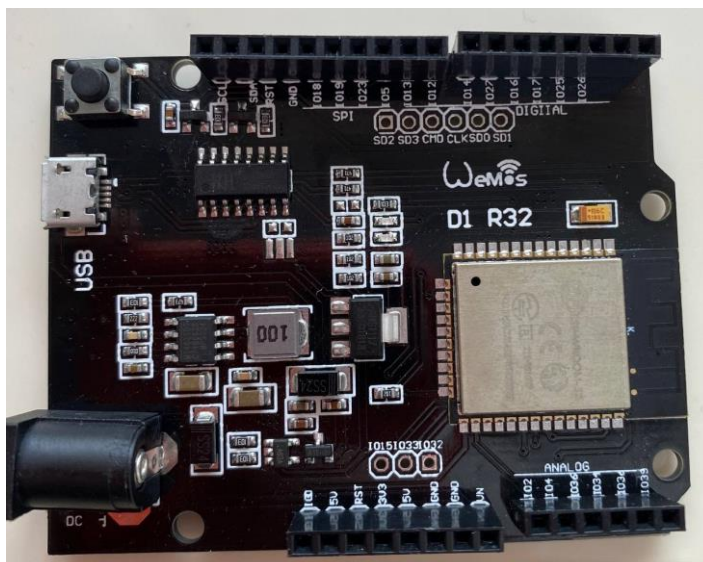


Рис. 9 – Мікрокомп'ютер ESP-32 Wemos D1 Mini

Характеристики ESP-32 [12]

CPU: Xtensa Dual-Core 32-bit LX6, 160 MHz или 240 MHz (до 600 DMIPS)

Memory: 520 KByte SRAM, 448 KByte ROM

Flash на модуле: 1, 2, 4... 64 Мб

Wireless:

- Wi-Fi: 802.11b/g/n/e/i, до 150 Mbps с HT40
- Bluetooth: v4.2 BR/EDR и BLE



3.4 Можливості ESP-32

В цьому розділі хотілося б звернути увагу не скільки на переваги цієї серії, скільки на можливості, потенціал в подальшому часі. Цей пристрій для створення програмних комплексів також має відмінну масштабованість, а також не має залежності від виробника, який створює комплектуючі пристрої або маленького спектру використання.

Це дозволяє використовувати такі мікроконтролери в будь-якій сфері, в системах розумного будинку чи в лабораторіях, в школах чи в медичній сфері.

Завдяки можливості передачі інформації за допомогою Wifi, про що було наголошено раніше, ми маємо можливість покращити використання будь-якої створеної програми, адже можна розробити свій інтерфейс, з яким використання, особливо людині, не знайомій з розробленням програми, буде набагато простіше.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		26

РОЗДІЛ 4

ВИБІР СЕРЕДОВИЩА РОЗРОБКИ

Перш за все, хотів наголосити на програмі, яка допоможе налаштувати мікроконтролер. Вона допоможе запрограмувати нашу мозкову частину заряджувального комплексу.

4.1 Аналоги до Arduino IDE

Для розробки ми обрали програму Arduino IDE. Але в цьому розділі хотів би представити аналоги до тої програми, яку ми вибрали:

- Microsoft Visual Studio. Інтегрована середовище розробки від компанії Microsoft.

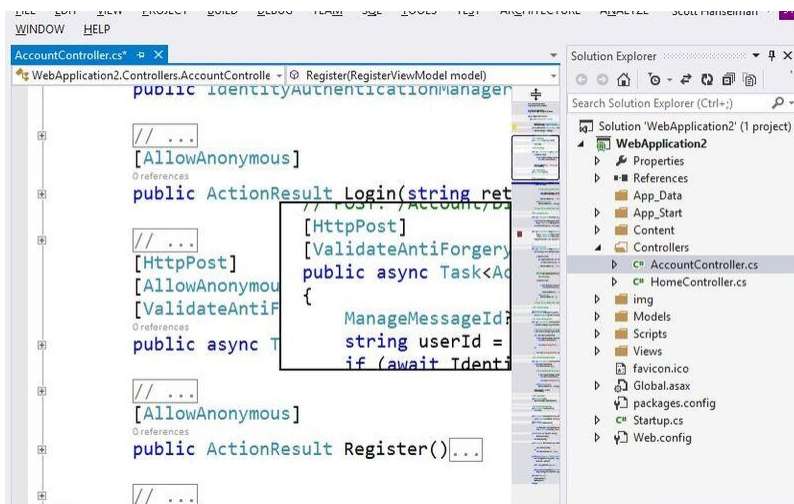


Рис. 10 - Інтерфейс програми Microsoft Visual Studio

Переваги цієї програми в тому, що в ній можна зробити власний чи керований код. Також включає в себе редактор коду, що підтримує IntelliSense, і рефакторинг програмного коду. В Microsoft Visual Studio вже є інтегровані інструменти, наприклад, конструктор форм чи створення додатків із графічним інтерфейсом.

- PlatformIO IDE

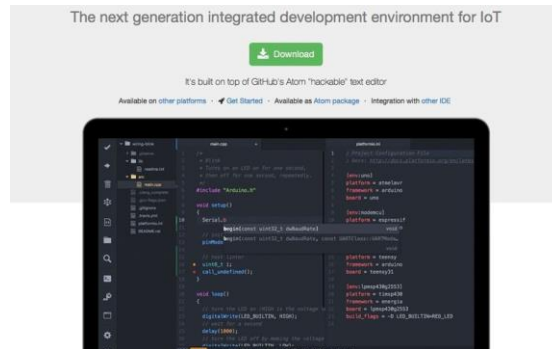


Рис. 11 – Інтерфейс програми PlatformIO IDE

Це середовище розробки, система складання без зовнішніх залежностей від програмного забезпечення. Код пишеться на C/C++. Перевагою цієї програми є наявність інтегрованого терміналу з інструментом PlatformIO CLI і з наявність потужного монітору послідовного порту.

4.2 Вибір програми для розробки

Було обрано програму Arduino IDE. Середу її розробки можна вільно завантажити з офіційного сайту [13]:



Рис. 11 – Середовище розробки Arduino

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		28

Незважаючи на очевидні обмеження порівняно з професійним середовищем розробки, Arduino IDE покриває 90% того, що потрібно досягти для аматорських проектів.

У мережі також є достатня кількість статей на тему встановлення та налаштування Arduino IDE для програмування модулів ESP32.

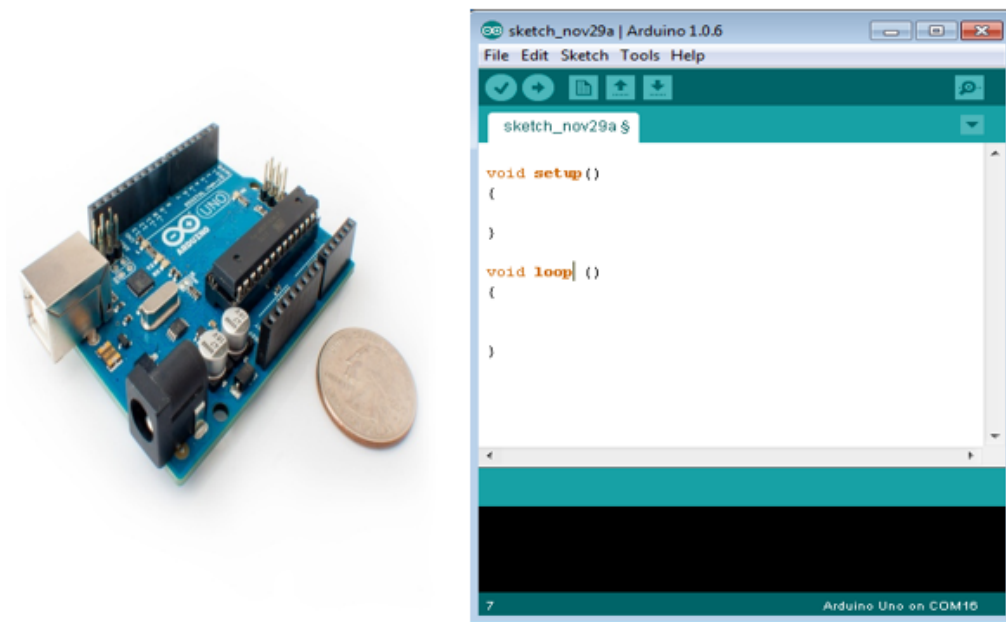


Рис. 12 – Приклад скетча в середовищі розробки

Алгоритм роботи пишеться на мові C++, після чого написаний код компілюється та завантажується у плату. Компіляція потрібна для виявлення помилок ще на стадії розробки.

4.3 Базовий інтерфейс програми Arduino IDE

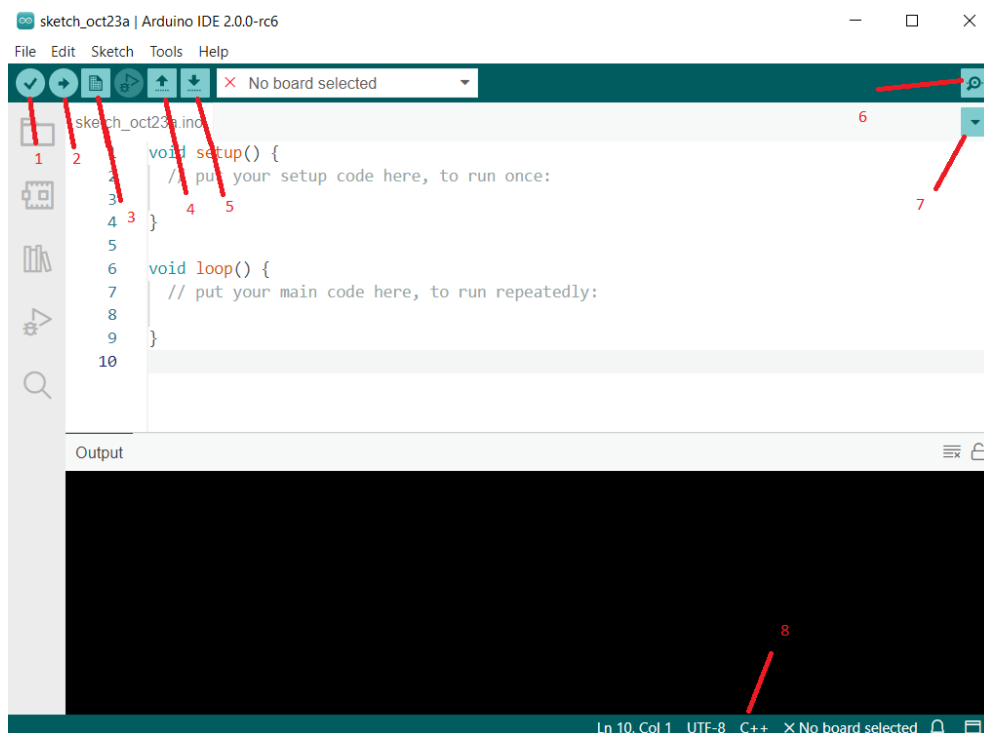


Рис. 13 - Базовий інтерфейс програми:

1. «Перевірити»
2. «Завантажити»
3. «Створити»
4. «Відкрити»
5. «Зберегти»
6. «Монітор порту»
7. Меню вкладок
8. Конфігурація обладнання

РОЗДІЛ 5

ПРОГРАМНИЙ КОД

1.1 Вихідні дані, налаштування початкового стану [14]

```
typedef enum ChargingState {  
    STATE_A, // Початкова напруга 12V, Not connected,  
             // ADC: 3950  
    STATE_B, // Початкова напруга 9V, Connected,  
             // ADC: 3408  
    STATE_C, // Початкова напруга 6V, Charging,  
             // ADC: 2847  
    STATE_D, // Початкова напруга 3V, Ventilation r  
             // ADC: 2390  
    STATE_E, // Початкова напруга 0V, No Power  
             // ADC: 1941  
    STATE_F, // Початкова напруга -12V, EVSE Error  
             // ADC: <1500  
    STATE_CUSTOM_OFF, // налаштований стан  
                     // для домашньої автоматизації,  
                     // без стягнення плати до STATE_CUSTOM_ON or STATE_A  
    STATE_CUSTOM_ON  
};
```

Рис. 14 – Функція початкового стану

Ключове слово **typedef** дозволяє створити псевдонім для будь-якого типу даних та використовувати його замість фактичного імені типу.

```
ChargingState pilotState = STATE_F;
```

Зазначена вище умова встановлює те, що при виявленні помилки стан зміниться на «без напруги», оскільки під час запуску не буде виявлено живлення.

GPIO, до якого підключено DS18B20

```
const int oneWireBus = 15;
```

Налаштовує приклад oneWire для зв'язку з будь-якими пристроями OneWire;

OneWire - двонаправлена шина зв'язку для пристроїв з низько-швидкісною передачею даних, в якій дані передаються ланцюгом живлення.

```
OneWire oneWire(oneWireBus);
```

Передає посилення OneWire на датчик температури Dallas;

```
DallasTemperature sensors(&oneWire);  
float temperatureC = 0.00;
```

```
int chargingCurrent = 14;
```

Рис. 15 - Поточний AMPS для зарядки

Схема PilotPin підключена до GPIO 34 (аналоговий ADC1_CH6);

```
int pilotPin = 34;
```

```
float ampValue = 0.00;  
String message = "";  
String amperes1 = "14";  
int minvaluePilot = 0;
```

Рис. 16 - Поточні значення датчика

```
int pilotValue = 0;  
float voltage = 0.00;  
String stateStr = "";
```

Рис. 17 - Змінна для зберігання початкового значення;

```
const char *ssidArray[] = {  
    "Your_SSID_HERE", "Your_2nd_SSID_HERE" ,  
    "Your_Telephone_Hot_Spot_SSID"  
};
```

Рис. 18 - Wi-fi налаштування


```
const char *passwordArray[] = {
    "YOUR_PASSWORD_HERE", "YOUR_2nd_PASSWORD_HERE" ,
    "Your_Telephone_Hot_Spot_Password"
};
```

Рис. 19 - Необов'язкові налаштування

```
const char* ssid = "";
const char* password = "";
```

Рис. 20 – данні, які заповнюються під час знаходження мережі

```
AsyncWebServer server(80);
AsyncWebSocket ws("/ws");
AsyncWebsocketClient * globalClient = NULL;
```

Рис. 21 - код, який створює асинхронний сервер на порту 80
(AsyncWebServer);

```
JSONVar sliderValues, internValues;
```

Рис. 22 - Змінна Json для зберігання значень повзунка;

```
String getSliderValues(){
    sliderValues["amperes1"] = String(amperes1);
    sliderValues["temperatureValue1"] = String(temperatureC);
    sliderValues["chargingState"] = String(stateStr);
    sliderValues["chargingAmps"] = String(ampValue);

    String jsonString = JSON.stringify(sliderValues);
    return jsonString;
}
```

Рис. 23 - Функція для отримання значень повзунка

Нам не слід з періодичністю оновлювати значення повзунка тому, що повзунок оновлюється користувачем на веб-сторінці, інші користувачі отримують сповіщення з допомогою попередньої функції getSliderValues().

Наступна функція оновлює лише згенеровані значення, наприклад, от ампер, які споживає автомобіль, а також значення внутрішньої температури зарядного пристрою і стан цього пристрою:

```
String sendSocketUpdates(){  
    internValues["temperatureValue1"] = String(temperatureC);  
    internValues["chargingState"] = String(stateStr);  
    internValues["chargingAmps"] = String(ampValue);  
  
    String jsonString = JSON.stringify(internValues);  
    return jsonString;  
}
```

Рис. 24 – функція для отправки веб-сокетів

Взагалі, слід використовувати "unsigned long" для змінних, які зберігають час;

```
unsigned long previousMillis = 0;
```

Ця змінна буде зберігати час останнього оновлення запиту, а константи змінюватися не будуть:

```
const long interval = 4000;
```

Значення змінної це інтервал посилення оновлень websocket клієнту, виражене в мілісекундах.

5.2 Розробка інтерфейсу програми

Нижче буде представлено вставку HTML коду, інтерфейс користувача на Android пристрої:

```
const char index_html[] PROGMEM = R"rawliteral(  
<!DOCTYPE html>  
<html>  
<head>  
<title>Charger Web Server</title>  
<meta name="viewport" content="width=device-width, initial-scale=1">
```

Рис. 25 - Мета інформація html файлу

Приведені стилі інтерфейсу (CSS код). Тут вказані основні розміри тексту, відступи між елементами, кольори тексту і фону, ширина та висота елементів, а також загальне centruвання :

```
<style>  
html {font-family: Arial; display: inline-block; text-align: center;}  
h2 {font-size: 2.3rem;}  
p {font-size: 1.9rem;}  
body {max-width: 500px; margin:0px auto; padding-bottom: 20px;}  
.slider { -webkit-appearance: none; margin: 14px; width: 380px; height: 25px; background: #FFD65C;  
outline: none; -webkit-transition: .2s; transition: opacity .2s;}  
.slider::-webkit-slider-thumb {-webkit-appearance: none; appearance: none; width: 35px; height: 35px;  
.slider::-moz-range-thumb { width: 35px; height: 35px; background: #003249; cursor: pointer; }  
</style>
```

Рис. 26 – CSS код стилів програми

Нижче представлена загальна інформація про температуру, максимально можливе зарядження в амперах, а також сталі величини, можливість включили або виключили зарядку.

					7.141.170033.ПЗ	Арк.
						35
Изм.	Лист	№ докум.	Підпис	Дата		

```

<body>
<h1>Choose Max Charging Current</h1>

<div class="slidecontainer">
  <h2 style="float: left;" >6 </h2> <h2 style="float: right;" > 32</h2>
  <input type="range" oninput="updateSlider(this)" onchange="updateSliderAMP(this)">
  <p>Max charging at: <span id="amperes1"></span> Amp.</p>
  <p></p>
  <p>Temperature at: <span id="tempValue"></span> &deg;C</p>
  <p>Charging State: </p>
  <p><span id="chargingText"></span></p>
  <p>Car Charges At <span id="chargingAtAmpsText"></span> A Now</p>
  <input type="button" id="btnChargeOn" value="Charger OFF" style="float: left;" onClick="sendOff();">
  <input type="button" id="btnChargeOn" value="Charger ON" style="float: right;" onClick="sendOn();">
</div>

```

Рис. 27 - HTML код, скелет інтерфейсу для відображення користувачу

На даний момент наша програма має такий вигляд (Рис.28):

Choose Max Charging Current

6 32

Max charging at: Amp.

Temperature at: °C

Charging State:

Car Charges At A Now

Charger OFF Charger ON

Рис. 28 – вигляд інтерфейсу програми

На цьому прикладі бачимо, що вже можна вибрати максимальну зарядку в амперах, при цьому нам доступні температурні показники.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		36

5.3 Програмування проекту під час виконання змін інтерфейсу

Тепер треба розробити частину JavaScript коду, яка відповідає за програмну частину, наприклад, під час зміни повзунка зарядки, відкривання чи закривання підключення тощо. На цьому етапі був використаний звичайний JS.

```
<script>
var gateway = `ws://${window.location.hostname}/ws`;
var websocket;
window.addEventListener('load', onload);
websocket.send("getValues");

function onload(event) {
  |   initWebSocket();
}

function getValues(){
  |   websocket.send("getValues");
}

function sendOff(){
  |   websocket.send("setStateCustomOff");
}

function sendOn(){
  |   websocket.send("setStateCustomOn");
}
```

Рис. 29 – Додавання слухачів подій за допомогою JavaScript

Ми додаємо «слухачів подій на кнопки Charger OFF та Charger ON, для того, щоб при натисканні на них відбувалась потрібна нам логіка програми.

```

function initWebSocket() {
    console.log('Trying to open a WebSocket connection');
    websocket = new WebSocket(gateway);
    websocket.onopen = onOpen;
    websocket.onclose = onClose;
    websocket.onmessage = onMessage;
}

function onOpen(event) {
    console.log('Connection opened');
    getValues();
}

function onClose(event) {
    console.log('Connection closed');
    setTimeout(initWebSocket, 2000);
}

```

Рис. 30 – функції для виконання під час відкриття чи закриття підключення

Тепер робимо логіку інпута (повзунку зарядки)

```

function updateSlider(element) {
    var sliderNumber = element.id.charAt(element.id.length-1);
    var sliderValue = document.getElementById(element.id).value;
    document.getElementById("amperes"+sliderNumber).innerHTML = sliderValue;
}

function updateSliderAMP(element) {
    var sliderNumber = element.id.charAt(element.id.length-1);
    var sliderValue = document.getElementById(element.id).value;
    document.getElementById("amperes"+sliderNumber).innerHTML = sliderValue;
    console.log(sliderValue);
    websocket.send(sliderNumber+"s"+sliderValue.toString());
}

```

Рис. 31 – функції для виконання під час зміни повзунка

Залишилося зробити функцію для Веб-сокету з певними умовами

(Рис. 32):

```
function onMessage(event) {  
    console.log(event.data);  
    var myObj = JSON.parse(event.data);  
    var keys = Object.keys(myObj);  
  
    for (var i = 0; i <= (keys.length - 1); i++){  
        var key = keys[i];  
  
        if (key == "temperatureValue1") {  
            document.getElementById("tempValue").innerHTML = myObj[key];  
        }  
        else if (key == "chargingState") {  
            document.getElementById("chargingText").innerHTML = myObj[key];  
        }  
        else if (key == "chargingAmps") {  
            document.getElementById("chargingAtAmpsText").innerHTML = myObj[key];  
        }  
        else {  
            document.getElementById(key).innerHTML = myObj[key];  
            document.getElementById("ampRange"+ (i+1)).value = myObj[key];  
        }  
    }  
}  
</script>  
)rawliteral";
```

Рис. 32 – функція для websocket

Ця функція (Рис. 33) зроблена, щоб замінювати заповнювач розділом кнопки на веб-сторінці, але код нічого не робить у програмі зарядного пристрою, тому його потрібно просто очистити:

```
String processor(const String& var){  
    //Serial.println(var);  
    if(var == "BUTTONPLACEHOLDER"){  
        String buttons = "";  
        return buttons;  
    }  
    return String();  
}
```

Рис. 33 – функція для заміни заповнювача розділом кнопки

```
#define OLED_RESET    27 // збрасує pin # (чи -1 якщо Arduino збрасує pin)
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
```

Рис. 34 - Відображення для дисплея SSD1306, підключеного до I2C
(контакти SDA, SCL)

```
3v3 to opamp
| | | 3.3V--|*| -- GND
Jtag Reset--|*| -- GPIO_23
GPIO_36--|*| OLED*| -- GPIO_22 <---> OLED SCL
GPIO_39--|*| -- GPIO__1
GPIO_34--|*Analog IN|*| -- GPIO__3
GPIO_35--|*Current sens OLED*| -- GPIO_21 <---> OLED SDA
GPIO_32--|*PWM 1kHz|*| -- NC / GND
GPIO_33--|*|*| -- GPIO_19
GPIO_25--|*230V Relay|*| -- GPIO_18
GPIO_26--|*|*| -- GPIO__5
GPIO_27--|*OLED_RESET|*| -- GPIO_17
GPIO_14--|*|*| -- GPIO_16
GPIO_12--|*|*| -- GPIO__4
| GND---|*|*| -- GPIO__0
GPIO_13--|*|*| -- GPIO__2
GPIO__9--|*|Temp Sensor*| -- GPIO_15 <---> Dallas DS18B20 Temperature sensor
GPIO_10--|*|*| -- GPIO__8
GPIO_11--|*|*| -- GPIO__7
| 5.0V--|*| EN USB BOOT *| -- GPIO__6
| | | |**_|**|
```

Рис. 35 - Розпи́нання мікроконтролера ESP32-WROOM-32

Також ми не використовували контакти ADC2 для вимірювання під час використання Wi-fi через те, що Wi-fi викликає переривання на контактах analog_in.

Ми використовували GPIO 32, 33, 34, 35, 36, 39 лише для аналогових вимірювань.

State:	Pilot Voltage:	EV Resistance:	Description:	Analog theoretic: (if pwm is on 1khz)
State A	12V	N/A	Not Connected	3.177 V = 3943 of 4096 on ADC
State B	9V	2.7K	Connected	2.811 V = 3489 of 4096 on ADC
State C	6V	882 Ohm	Charging	2.445 V = 3034 of 4096 on ADC
State D	3V	246 Ohm	Ventilation Required	2.079 V = 2580 of 4096 on ADC
State E	0V	N/A	No Power	1.713 V = 2126 of 4096 on ADC
State F	-12V	N/A	EVSE Error	249.198 mV = 309 of 4096 on ADC

Рис. 36 - Стан системи зарядження

Початкову напругу можна таким чином:

1) Спочатку треба взяти резистор із табл. опорів EV, потім помістили його між першим контактом і землею.

2) Початкова напруга впаде до 9 В із резистором 2.7 К, тоді можна буде виміряти АЦП.

Краще було тестувати напругу с 100% ШІМ від операційного підсилювача, а потім за допомогою функції analogRead();

3) На виході PilotPin покаже нам бажане значення АЦП.

Також є варіант передачі різних напруг від джерела живлення через контакт PP з землею. (але на ланцюзі має бути напруга зміщення, бо контакти ESP можуть бути пошкоджені).

```

State A: 3.19V and ADC: 3950
State B: 2.75V and ADC: 3408
State C: 2.30V and ADC: 2847
State D: 1.92V and ADC: 2390
State E: 1.57V and ADC: 1941
State F:

```

Рис. 37 - Вимірний аналог

На рис. 38 представлений розрахунок ШІМ-сигналу для заряджання @ x AMPS: (дійсний для до 51 A):

$$\text{AMPS} = \text{Duty cycle} \times 0.6 \text{ (duty cycle is in \%)}$$

$$\text{Duty Cycle} = \text{AMPS} / 0.6$$

як приклад: $6A / 0.6 = 10\% \text{ PWM}$

$$16A / 0.6 = 26.666\% \text{ PWM}$$

$$10\% \text{ PWM} \times 0.6 = 6A$$

$$20\% \text{ PWM} \times 0.6 = 12A$$

$$51-80A, \text{ AMPS} = (\text{Duty Cycle} - 64)2.5$$

Рис. 38 – Розрахунок ШІМ-сигналу

З розрахунком ШІМ-сигналу детальніше можна ознайомитися по посиланню за посиланням [28]

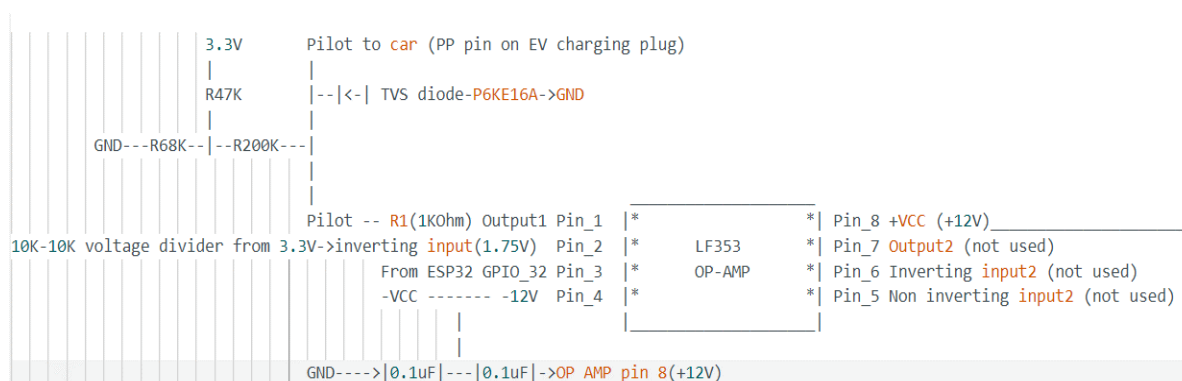


Рис. 39 - Схема операційного підсилювача

Хотів би зазначити, що перевірки безпеки не включені в цю версію схеми, доказ концепції, лише якщо ви не підключені до мережі, захищеної пристрій захисного відключення (ПЗВ). Перевірка діодів теж не була реалізована.

Але було реалізовано СТ- для вимірювання струму 32 A на 2000 обертових струмових кліщах, резистор 68 Ом забезпечує діапазон вимірювання до 34 A.


```
float getChargingCurrent(int pin, int noSamples){
    int analog = checkAnalog(pin, noSamples);

    double amps = (0.0138 * analog) - 26;

    return amps;
}
```

Рис. 41 – Функція для отримання поточного значення струму

На рис. 41 представлена формула з експериментальних значень на 243 В змінного струму перевіреної лінії тестування на резистивних нагрівачах 600-1500 Вт.

Також був використаний датчик струму, від sparkfun 30А, як за посиланням [31], а також резистор навантаження 68 Ом.

```
void scanNet(){
    Serial.println("scan start");

    // WiFi.scanNetworks поверне нам кількість знайдених мереж
    int n = WiFi.scanNetworks();
    int y = sizeof(ssidArray) / sizeof(ssidArray[0]);
    Serial.println("scan done");
    if (n == 0) {
        Serial.println("no networks found");
    } else {
        Serial.print(n);
        Serial.println(" networks found");
        for (int i = 0; i < n; ++i) {
            // Print SSID та RSSI для кожної знайденої мережі
            Serial.print(i + 1);
            Serial.print(": ");
            Serial.print(WiFi.SSID(i));
            Serial.print(" (");
            Serial.print(WiFi.RSSI(i));
            Serial.print(")");
            Serial.println((WiFi.encryptionType(i) == WIFI_AUTH_OPEN)? " ":"*");

            for (int x = 0; x < y; x++){
                if (WiFi.SSID(i) == ssidArray[x]){ // Ми знайшли мережу, що відповідає мережі з нашого списку
                    ssid = ssidArray[x];
                    password = passwordArray[x];
                    Serial.print("connect to network: ");
                    Serial.println(ssidArray[x]);
                    i = n; // робота циклу виконана, ми підключаємось до першої мережі та виходимо...
                }
            }
            delay(10);
        }
    }
    Serial.println("");
}
```

Рис. 42 - Функція для знаходження потрібної Wi-fi мережі

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		44

```

void initWebSocket() {
    ws.onEvent(onEvent);
    server.addHandler(&ws);
}

int checkAnalog(int analogPinToTest, int noSamples){
    int maximum = 0;
    int minimum = 5000;
    int value;
    for (int i = 0; i <= noSamples; i++) {
        value = analogRead(analogPinToTest);
        if (value <= minimum){
            minimum = value;
            minvaluePilot = minimum;
        }
        if (value >= maximum){
            maximum = value;
        }
    }

    return maximum;
}

```

Рис. 43 – Функція для перевірки аналогового сигналу

```

int chargingPWM(int ampsToConvert){
    float pwmsignal = (ampsToConvert + 3)*17.06667;
    return (round(pwmsignal)-1);
}

```

Рис. 44 – Функція для встановлення ШІМ-сигналу (PWM)

$\text{float pwmsignal} = \text{ampsToConvert} / 0,05859375;$

0,05859375 дорівнює $1/1024$ від 1 А при використанні роздільної здатності в 10 біт.

Також це $1/1024$ від 1А при використанні 10-бітної роздільної здатності. В нашому випадку вийшло так, що наш електромобіль робить неправильний крок на 3 А за пілотним сигналом ШІМ і починає заряджатися лише тоді, коли ми робимо запит в 9А, а потім заряджає 6А.

```

void onEvent(AsyncWebSocket *server,
AsyncWebSocketClient *client,
AwsEventType type, void *arg, uint8_t *data,
size_t len) {
    switch (type) {
        case WS_EVT_CONNECT:
            Serial.printf("WebSocket client #%u connected from %s\n",
            client->id(),
            client->remoteIP().toString().c_str());
            globalClient = client;
            break;
        case WS_EVT_DISCONNECT:
            Serial.printf("WebSocket client #%u disconnected\n",
            client->id());
            globalClient = NULL;
            break;
        case WS_EVT_DATA:
            handleWebSocketMessage(arg, data, len);
            break;
        case WS_EVT_PONG:
        case WS_EVT_ERROR:
            break;
    }
}

```

Рис. 45 – Функція для зміни стану сервера

Вище використовується функція з допомогою switch...case, де в залежності від стану сервера використовується своя логіка, а далі йде функція для сповіщення користувачеві.

```

void handleWebSocketMessage(void *arg, uint8_t *data, size_t len) {
    AwsFrameInfo *info = (AwsFrameInfo*)arg;
    if (info->final && info->index == 0 &&
        info->len == len &&
        info->opcode == WS_TEXT) {
        data[len] = 0;
        message = (char*)data;
        if (message.indexOf("1s") >= 0) {
            amperes1 = message.substring(2);
            chargingCurrent = amperes1.toInt();
            ledcWrite(PWM1_Ch, chargingPWM(chargingCurrent));
            Serial.println(chargingCurrent);
            Serial.print(getSliderValues());
            notifyClients(getSliderValues());
        }
    }
}

```

Рис. 46 – Функція при зміні стану WebSocket

Далі на рис. 47 робиться перелік умов для більш коректного сповіщення для користувача програми:

```

if (message.indexOf("2s") >= 0) {
    sliderValue2 = message.substring(2);
    dutyCycle2 = map(sliderValue2.toInt(), 0, 100, 0, 255);
    Serial.println(dutyCycle2);
    Serial.print(getSliderValues());
    notifyClients(getSliderValues());
}
if (message.indexOf("3s") >= 0) {
    sliderValue3 = message.substring(2);
    dutyCycle3 = map(sliderValue3.toInt(), 0, 100, 0, 255);
    Serial.println(dutyCycle3);
    Serial.print(getSliderValues());
    notifyClients(getSliderValues());
}
*/
if (strcmp((char*)data, "getValues") == 0) {
    notifyClients(getSliderValues());
}
if (strcmp((char*)data, "setStateCustomOff") == 0) {
    // використання на веб-сторінці за допомогою javascript:
    // websocket.send("setStateCustomOff");
    pilotState=STATE_CUSTOM_OFF;
    notifyClients(getSliderValues());
    Serial.println("State: STATE_CUSTOM_OFF");
}
}

```

Рис. 47 – Перелік умов при зміні сповіщення

Наступна функція допоможе нам у випадку, якщо електромобіль вже заряджається, а ми не хочемо змінювати стан на який-завгодно, крім вимкненого.

Тому ми встановили ШІМ на значення 1023 (PWM1_DutyCycle = 1023).

У випадку, якщо цього не робити, автомобіль припинить свою зарядку та спробує запустити її знову, що призведе до помилки заряджання.

Тому було вирішено встановити значення на постійну величину (Рис. 48):

```
if (strcmp((char*)data, "setStateCustomOn") == 0) {  
    if (!(pilotState == STATE_C)){  
        pilotState=STATE_CUSTOM_ON;  
        digitalWrite(RELAY_PIN, LOW);  
        // початок з микнутим реле  
        PWM1_DutyCycle = 1023;  
        // вмикаємо ШІМ на постійне значення, +12 В на старті,  
        // щоб ми не отримувати код ПОМИЛКИ EVSE (STATE_F)  
        ledcWrite(PWM1_Ch, PWM1_DutyCycle);  
        notifyClients(getSliderValues());  
        Serial.println("State: STATE_CUSTOM_ON");  
    }  
}  
}
```

Рис. 48 – Функція для встановлення значення


```

void notifyClients(String sliderValues) {
    ws.textAll(sliderValues);
}

void printDisplayData(){
    display.clearDisplay();
    display.setTextSize(1);
    // Встановлюємо нормальне відношення тексту 1 до 1
    display.setTextColor(WHITE);
    // Відображаємо білий текст
    display.setCursor(0,2);
    display.print("SSID: ");
    display.println(ssid);
    display.println("Http://charger.local");
    display.print("charging at ");
    display.println(ampValue);
    display.print("SSR temp is: ");
    display.println(temperatureC);
    //display.print("Voltage is: ");
    //voltage = (3.3 / 4096) * pilotValue;
    //display.println(voltage);
    display.print("Setpoint at ");
    display.print(chargingCurrent);
    display.println(" Amps");
    display.println(stateStr);
    // відображає статус на скріншоті
    display.println(WiFi.localIP());
    display.display();
    //delay(2);
}

```

Рис. 49 – Функція встановлення інтерфейсу програми

На рис. 49 було встановлено відношення тексту в масштабі 1 до 1, а також відображено білий текст та статус на скріншоті.

На рис. 50 представлена функція для перевірки стану зарядки, в залежності від значення аналогово-цифрового перетворювача:

```
void CheckState(ChargingState oldChargingState,
int adc_value){
    ChargingState newState;
    int ampsPWM = chargingPWM(chargingCurrent);
    Serial.print("pwm for charging = ");
    Serial.println(ampsPWM);

    if (adc_value >= 3779 && adc_value < 4096){
        newState = STATE_A;
    }
    else if (adc_value >= 3150 &&
adc_value < 3779){
        newState = STATE_B;
    }
    else if (adc_value >= 2618 &&
adc_value < 3150){
        newState = STATE_C;
    }
    else if (adc_value >= 2166 &&
adc_value < 2618){
        newState = STATE_D;
    }
    else if (adc_value >= 1700 &&
adc_value < 2166){
        newState = STATE_E;
    }
    else if (adc_value >= 0 &&
adc_value < 1700){
        newState = STATE_F;
    }
}
```

Рис. 50 – Функція для перевірки стану зарядки

					7.141.170033.ПЗ	Арк.
						50
Изм.	Лист	№ докум.	Підпис	Дата		

Тепер в залежності від стану будемо робити наступні дії, описані на рис. 51 (а, б, в) за допомогою коментарів:

```
if (!(oldChargingState == newState)){
    pilotState = newState;
    switch (pilotState){
        case STATE_A:
            stateStr = "Not Connected";
            digitalWrite(RELAY_PIN, LOW);
            PWM1_DutyCycle = 1023;
            // Включаємо ШІМ, поставивши незмінну величину
            ledcWrite(PWM1_Ch, PWM1_DutyCycle);
            //digitalWrite(PILOT_GPIO, HIGH);
            // ставимо +12V DC на початку
            break;
        case STATE_B:
            stateStr = "Connected";
            //digitalWrite(PILOT_GPIO, LOW); // выключаємо DC напругу
            digitalWrite(RELAY_PIN, LOW);
            // Чекаємо, пока EV перейде в режим заряджання
            PWM1_DutyCycle = ampsPWM; // % если 1023 max = Square wave
            ledcWrite(PWM1_Ch, PWM1_DutyCycle);
            Serial.println(ampsPWM);
            delay(20);
            break;
```

Рис. 51, а – Функція перевірки за допомогою стану зарядження

```
        case STATE_C:
            //PWM1_DutyCycle = 0; // выключаємо ШІМ
            //ledcWrite(PWM1_Ch, PWM1_DutyCycle);
            //delay(100);
            // Оголошуємо місткість заряджання
            PWM1_DutyCycle = ampsPWM; // % of 1023 max =14A
            ledcWrite(PWM1_Ch, PWM1_DutyCycle);
            delay(10); // імітуємо час замикання реле
            digitalWrite(RELAY_PIN, HIGH);
            stateStr = "Charging";
            break;
        case STATE_D:
            stateStr = "Ventilation Required";
            digitalWrite(RELAY_PIN, LOW); // без зарядки
            // Оголошуємо місткість заряджання
            PWM1_DutyCycle = ampsPWM; // выключаємо ШІМ
            ledcWrite(PWM1_Ch, PWM1_DutyCycle);
            break;
        case STATE_E:
            stateStr = "No POWER,";
            stateStr += '\n';
            stateStr += " Ready to connect";
            digitalWrite(RELAY_PIN, LOW); // без зарядки
            PWM1_DutyCycle = 1023; // ставимо +12V DC на початку
            ledcWrite(PWM1_Ch, PWM1_DutyCycle);
            break;
```

Рис. 51, б – друга частина станів зарядження

```

case STATE_F:
    stateStr = "--- EVSE ERROR ---";
    digitalWrite(RELAY_PIN, LOW); // без зарядки
    // Оголошуємо місткість заряджання
    PWM1_DutyCycle = 0; // виключаємо ШІМ
    ledcWrite(PWM1_Ch, PWM1_DutyCycle);
    break;
}
}
printDisplayData();
}

```

Рис. 51, в – останній стан системи зарядження

```

void setup() {
    Serial.begin(115200);

    if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3D)) {
        // Передаємо 0x3D для 128x64px
        Serial.println(F("SSD1306 allocation failed"));
        for(;;);
    }
}

```

Рис. 52 – Функція для генерації напруги дисплея

SSD1306_SWITCHCAPVCC – генерує напругу дисплея від внутрішньої напруги 3,3 В.

Вираз for(;;) не дає циклу повторюватися нескінченну кількість разів.

```
sensors.begin();
```

Цей вираз запустить датчик DS18B20;

```

display.clearDisplay();
display.println("Bil-lader V_1");
display.display();
delay(500);
display.clearDisplay();

```

Рис. 53 – Вираз для чистки буфера обміну

На рис. 54 представлено відображення растрового зображення відбувається протягом 2 секунд:

```
display.drawBitmap(0, 0,  
FluidSensorFluidSensorTechnology,  
| 128, 64, WHITE);  
display.display();  
//delay(2000); // пошук Wi-Fi займає час  
display.clearDisplay();
```

Рис. 54 – Вираз для встановлення зображення

На цьому етапі рекомендується встановити Wi-фі в режим станції і відключитись від точки доступу, якщо вона була підключена до пристрою (Рис. 55):

```
WiFi.mode(WIFI_STA);  
WiFi.disconnect();  
delay(100);  
  
Serial.println("finding network to connect to...");  
scanNet();  
  
WiFi.mode(WIFI_AP);
```

Рис. 55 – Встановлення Wi-фі мережі в режим станції

```

WiFi.begin(ssid, password);

while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi..");
}

if(!MDNS.begin("Charger")) {
    Serial.println("Error starting mDNS");
    return;
}

Serial.println(WiFi.localIP());

initWebSocket();

```

Рис. 56 - Налаштування для Wi-fi мережі

Додаємо нашу програму до MDNS-SD:

```

    MDNS.addService("http", "tcp", 80);

server.on("/", HTTP_GET,
[](AsyncWebServerRequest *request){
    request->send_P(200, "text/html", index_html, processor);
});

server.on("/hello", HTTP_GET,
[](AsyncWebServerRequest *request){
    request->send(200, "text/plain", "Hello World");
});

```

Рис. 57 - Налаштування маршруту до корінної або веб-сторінки

Після цього запускаємо наш сервер:

```

server.begin();
Serial.println("TCP server started");

```

Також треба ініціалізувати штифт ШІМ на визначеній частоті та роздільній здатності (Рис. 58):

```
pinMode(34, INPUT);
pinMode(RELAY_PIN, OUTPUT);
digitalWrite(RELAY_PIN, LOW);
ledcAttachPin(PILOT_GPIO, PWM1_Ch);
ledcSetup(PWM1_Ch, PWM1_Freq, PWM1_Res);
```

Рис. 58 – ініціалізація штифту ШІМ

Змінна записує значення початкової напруги :

```
pilotValue = checkAnalog(pilotPin, 400);
```

За допомогою виразу на рис. 59 відображаємо значення зарядного струму. Але тільки у випадку, коли зарядка йде, інакше буде відображати 0.00 А:

```
if (pilotState == STATE_C){
|   ampValue = getChargingCurrent(CURRENT_SENS_GPIO, 1000);
} else{
|   ampValue = 0.00;
}
```

Рис. 59 – Функція для отримання поточного значення струму

У випадку, коли домашня автоматизація вимкне наш зарядний пристрій, ми не змінюємо стан, як зображено на рис. 60:

```
if (pilotState == STATE_CUSTOM_OFF){
|   stateStr = "Charging on Schedule";
|   digitalWrite(RELAY_PIN, LOW); // без зарядки
|   // Оголошуємо місткість заряджання
|   PWM1_DutyCycle = 0; // виключаємо ШІМ
|   ledcWrite(PWM1_Ch, PWM1_DutyCycle);
} else {
|   checkState(pilotState, pilotValue);
}
```

Рис. 60 – умова при якій зарядний комплекс вимикається

Інакше стався б такий випадок, що ми або не отримали custom_off стан, або він не був реалізований за допомогою STATE_CUSTOM_OFF.

Тоді краще перевірити стан підключення зарядного пристрою та прослідкувати за вказівками самого електромобіля.

```
unsigned long currentMillis = millis();
if (currentMillis - previousMillis >= interval) {

    previousMillis = currentMillis;
    printDisplayData();
    // Відображаємо статус на OLED
    sensors.requestTemperatures();
    temperatureC = sensors.getTempCByIndex(0);
    if(globalClient != NULL && globalClient->status() == WS_CONNECTED){
        globalClient->text(sendSocketUpdates());
        // відправляємо дані в форматі JSON
    }
}
```

Рис. 61 – Функція для кешування даних

Лишилося тільки закешувати данні, тобто зберегти останній раз, коли ми надсилали повідомлення до веб-браузеру (Рис.61).

РОЗДІЛ 6

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

Охорона праці є невід'ємною частиною всього процесу, адже з цим питанням пов'язане життя та здоров'я працівників, а з ефективною та безпечною роботою на підприємстві укріплюється економіка держави, створюється її стабільність. Крім того, відсутність небезпечних випадків позначається і роботі в цілому, дає гарантії працівнику, а також скріплює весь колектив. Це в свою чергу впливає на продуктивність підприємства.

Дипломна робота має як теоретичний характер, так і практичний. По-перше, розробка апаратно-програмного комплексу зарядження електромобіля потребує роботи з комп'ютерною технікою, а тому слід дотримуватися вимог охорони праці на робочому місці, адже сприятливі умови при розробці програмного коду або іншої теоретичної роботи забезпечують не тільки правильність виконання, а і максимальну ефективність. Практична частина роботи буде складатись з вимірювань при напрузі за допомогою осцилографа. Безумовно, для безпечних вимірювань і відсутності травм вкрай необхідно буде перевірити стан робочого місця, а також дотримуватися вимог безпеки. Також було додано розділ з техніки безпеки при обслуговуванні акумуляторів.

6.1 Вимоги безпеки праці під час виконання робіт на робочому місці

Навіть при тестових вимірюваннях дотримування норм безпеки повинні бути в пріоритеті. Під час роботи можна зіткнутися з такими питаннями, перш за все пов'язаними з прямим контактом з обладнанням, яке знаходиться під напругою, а також робота з вимірюванням чи при написанні програмуванні програмного коду включає в себе: порушення електробезпеки, температурний режим, тобто мікроклімат, рівень освітлення, електромагнітне випромінювання, шум, вібрація. Також поговоримо про регламентований час згідно с нормами праці.

					7.141.170033.ПЗ	Арк.
						57
Изм.	Лист	№ докум.	Підпис	Дата		

Під час розробки програмного коду (теоретична робота), ми розглянемо організацію робочого місця, його ергономічне та естетичне оформлення, а також монотонність, малорухливість.

6.1.1 Електробезпека

Безпека по експлуатації електрообладнання складається із організаційних та технічних засобів та заходів, які забезпечують недопущення електротравматизму при роботі з електроустановками.

Згідно Правил улаштування електроустановок (ПУЕ), практична робота з електроприборами відноситься до роботи в пожежонебезпечній зоні.

Недотримання електробезпеки має більш небезпечний характер, тому апаратуру, в нашому випадку аналіз даних за допомогою осцилографа, слід застосовувати в пилонапроникному виконанні. Струмівідні частини повинні мати подвійну ізоляцію проводів згідно правила 1.7.87 ПУЕ [15]. Має місце бути недосяжність ненавмисного доторкання до струмоведучих частин. Також пожежонебезпечна зона повинна містити в собі захисне заземлення або занулення конструкцій, які можуть опинитися під напругою. Має бути захисне відключення, а також попереджувальні знаки безпеки (плакати, написи, тощо). Мають бути використані справні штепсельні з'єднання і електророзетки заводського виготовлення.

Поради які допоможуть звести ризик враження електричним током до мінімуму:

- Ізолювання проводів, а також інших струмопровідних частин.
- Працівник під час вимірювань чи при контакті з пристроями під напругою повинен стояти на спеціальному гумовому коврику.
- Електричне обладнання, вимірювальні апарати, електропроводи повинні відповідати правилам улаштування електроустановок (ПУЕ), тобто мати захист від струмів короткого замикання та аварійних режимів.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		58

- Небезпечні струмопровідні частини, які можуть бути на електроустановках, повинні бути недоступні для випадкового торкання, а відкриті провідні частини мають бути безпечними для нормальних умов.

Виконуючи роботу за комп'ютером, по-перше, треба слідкувати за сухістю рук, не допускати потрапляння рідинних речовин на ПК, його корпус чи допоміжні пристрої. Також не слід без потреби виключати або включати пристрій дуже часто. Під час протирання комп'ютера від пилу важливо вимкнути його, а також екран.

6.1.2 Мікроклімат та якість повітря

В робочому приміщенні мають дотримуватися санітарні норми мікроклімату відповідно до ДСН 3.3.6.042 – 99 [16]. Такий показник дає визначення категорії важкості роботи. Невідповідність нормам мікроклімату загрожує необхідності проведення відповідних заходів, які допоможуть нормалізувати параметри мікроклімату.

Розглянемо рівень мікроклімату за такими показниками:

- Температура поверхні
- Швидкість руху повітря
- Інтенсивність руху повітря
- Температура повітря
- Відносна вологість повітря

Норми параметрів мікроклімату для виробничих приміщень (відповідно до ДСН 3.3.6.042 – 99) у зимовий період:

					7.141.170033.ПЗ	Арк.
						59
Изм.	Лист	№ докум.	Підпис	Дата		

Період року	Категорія робіт	Температура повітря, °С	Відносна вологість, %	Швидкість руху, м/сек.
Холодний період	Середньої важкості II_a	19 – 21	40 – 60	0.2
	Середньої важкості II_b	17 – 19	40 – 60	0.2

Оптимальні умови мікроклімату для виробничого приміщення для категорії II_a та II_b середніх робіт

Є також ступені впливу на тепловий стан людини, це оптимальний ступінь та допустимий.

Для нормальних умов при виконанні роботи необхідна чистота повітря. Внаслідок вимірювальної діяльності у повітря можуть потрапити різноманітні шкідливі речовини. Вони можуть потрапити у організм людини за допомогою органів дихання, травлення або на шкіру. Можливе потрапляння маленьких частинок пилу, що негативно вплинути на легені працівника.

Шкідливість повітря може також призвести до захворювання легень. Для зменшення цієї проблеми в приміщенні для практичних робіт повинна бути вентиляція.

Іншим чинником є гранично допустимі концентрації (ГДК) запиленості повітря [17]. Це такі концентрації, які не викликають шкідливих наслідків при щоденній 8-годинній роботі протягом усього стажу (25 років). За такими показниками висота приміщення повинна становити 2,0 м. А якщо концентрація пилу перевищує допустимі норми, слід користуватись засобами індивідуального захисту. Це може бути респіратор, а також інший спеціальний одяг.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		60

У таблиці нижче представлена таблиця інших гранично допустимих концентрацій шкідливих речовин в повітрі робочої зони

Назва речовини	Гранично допустима концентрація, $мг/м^3$	Клас небезпеки
Пил рослинного і тваринного походження з домішками:		
Скляне волокно	3	4
Цемент, барит, штучні абразиви	5	4
Азбест	2	4
Вапно	6	4
оксиду кремнію (IV) менше 2 % (борошняна, бавовняна, цукрова, деревинна)	6	4
Граніт	2	4
оксиду кремнію (IV) понад 10 % (зернова, бавовняна, лляна, вовняна)	2	4
оксиду кремнію (IV) від 2 до 10 %	4	4
Тютюн	3	3
Вугілля:		
без оксиду кремнію (IV)	10	4
до 10 % оксиду кремнію (IV)	4	4
Інше, без домішок	10	4

Пил з вмістом понад 70 % вільного оксиду кремнію (IV) та його кристалічної модифікації (кварц, кварцит та інші)	1	3
-----------------------------------------------------------------------------------------------------------------	---	---

6.1.3. Освітлення

Розглянемо рівень робочого освітлення, воно має бути рівномірне. Показник цей є енергетичним, адже без нормального освітлення у людини може відбутися втома зорового аналізатора, що підвищує ризик травматизму чи недостатньої ефективності роботи.

При виконання теоретичної роботи очі повинні відчувати себе зручно, тобто не бути в затемненому освітленні чи в надто яскравому.

Також монітор краще не стави тильною стороною до вікна, бо при роботі денне світло буде спрямовано людині в очі, що також може призвести до їх швидкої перевтоми.

В умовах технічних робіт буває штучне, природне та комбіноване освітлення. В нашому випадку використовується штучне освітлення, але воно також може бути як загальне, так і комбіноване.

Згідно з ДБН В.2.5-28:2018 «Природне і штучне освітлення», НПАОП 0.00-7.15-18 [18] загальне освітлення це таке, при якому розміщення світла має бути не нижче 2.5 м над підлогою, а також рівномірно чи з урахуванням розташування зони вимірювання.

6.1.4. Електромагнітне випромінювання

Згідно ДСН 239-96 [19] через взаємодію із пристроєм для вимірювання електричного сигналу людина не тільки має ризик при роботі із напругою, а і може бути уражена електромагнітним випромінюванням.

					7.141.170033.ПЗ	Арк.
						62
Изм.	Лист	№ докум.	Підпис	Дата		

Взаємодія електромагнітного поля високої частоти при впливі на організм людини може спричинити підвищення кров'яного тиску, трофічні явища, наприклад, випадіння волосся чи ламкість нігтів. Електромагнітне поле викликає зміну поляризації атомів та молекул, що є складовою частиною клітин, це може викликати небезпечний нагрів. Для здійснення профілактичних заходів було встановлено гранично допустимий рівень (ГДР) електромагнітного поля для приміщень, які працюють в умовах частот (30 кГц – 300 ГГц). Перевищення допустимих норм може викликати професійні захворювання.

Але є засоби, які зменшують це випромінювання на людину. Наприклад, екранування, для яких використовується: мідь, сталь, латунь, а також алюміній та його сплави. Ці екрани виготовляють з сіток у вигляді камер замкнутого типу чи металевих листів, або кожухів, які мають бути під'єднані до системи заземлення. Інформацію знайдено згідно статі. [20]

6.1.5 Медичні заходи на підприємстві

Методико-профілактичні заходи включають в себе проведення періодичних медичних оглядів працівників. Цей захід направлений на виявлення симптомів хронічних чи професійної хвороби, а також в профілактичних цілях. Періодичність медичних оглядів залежить від умов праці. Важливо, що при виявленні хронічного захворювання працівник повинен бути взятий на відповідний облік.

6.1.6 Шум, вібрація

В цьому підрозділі буде приведено санітарні норми щодо виробничого шуму, ультразвуку, а також вібрації згідно ДСН 3.3.6.037-99 [21].

Наявність шуму під час виконання практичної роботи може викликати у працівника проблеми із слухом, а ультразвук хоч і не має впливу на слух, проте може викликати болі в голові та загальну втому.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		63

Також через ці звуки можуть з'явитися порушення нервової системи, а також вестибулярного апарату. Згідно санітарних норм, задля запобігання таких проблем, працівник має носити спеціальні навушники.

Вібрація ж, призводить більш важких проблем, це, наприклад, може призвести до розриву тканин, струсу мозку, порушення нервової системи і кровообігу. За наявності вібрації треба зробити віброгасіння, наприклад, ізоляція джерела коливань спеціальними віброізоляторами тощо.

6.1.7 Організація робочого місця

Розглянемо організаційні заходи та методико-профілактичні.

По-перше, у працівника, який працює за комп'ютером, повинен бути достатній простір на робочому місці для персонального комп'ютера, а також периферійних пристроїв (клавіатура, маніпулятор, модем тощо). Треба також враховувати відношення розміщення робочого місця до світлових прорізів, а також відстань між робочими столами, якщо працівників більше одного.

Треба також враховувати розташування монітора і клавіатури (принтера), по відношенню один до одного та то людини. Для комфортної праці та частої роботи відстань між монітором і очами повинна бути більше 0,4 м, але не більше 0,75 м.

6.1.8 Ергономічне та естетичне оформлення робочого місця

Розглянемо ергономічні та естетичні вимоги до робочого місця, а також проаналізуємо гігієнічну оцінку роботи працівника, проведемо заходи, що уникають прояву нервових захворювань і підвищують працездатність під час виконання магістерської роботи при програмуванні.

- Перш за все згідно пункту 2.2 ДСанПіН 3.3.2-007-98 [22] заборонено розміщувати робоче місце у підвальних чи цокольних поверхах.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		64

- Для комфорту працівника приміщення повинно бути обладнане системою опалення та кондиціонування повітря.
- Для мікрокліматичних умов і зберігання гігієни в приміщенні потрібно робити щоденне вологе прибирання.
- Приміщення має оснащувати аптечку першої медичної допомоги.
- Для уникнення нервових захворювань та правильного відпочинку працівника приміщення має бути обладнане побутовою кімнатою для відпочинку під час перерви від роботи, це можна психічно розвантажитись.

6.1.9 Монотонність, малорухливість

Згідно санітарним нормам при роботі з екранними пристроями [22] найбільш небезпечніші фактори при роботі з комп'ютерами під час виконання роботи це монотонність та малорухливість, адже майже увесь час працівник виконує однотипну роботу в сидячому положенні. Це має свій негативний вплив на зір людини, а якщо робота має високий ступінь відповідальності, це призводить ще й до нервового та емоційного виснаження. Також може виникнути розумове перенапруження через велику кількість різних складних завдань під час роботи.

Задля профілактики рівня нервової напруженості та інших негативних чинників на підприємстві мають проводитись медичні огляди (про них йде у підрозділі «Медичні заходи на підприємстві, підрозділ 4.1.5.»). Робоче місце працівника, виконуючого теоретичну роботу повинно мати такий розмір, щоб при цьому був простір у працівника для зміни положення тіла.

Є також санітарні вимоги до робочого крісла працівника, яке дасть змогу йому легко рухатись та мати зручне положення тіла. Також крісло має регулюватись, як по висоті, так і його спинка.

					7.141.170033.ПЗ	Арк.
						65
Изм.	Лист	№ докум.	Підпис	Дата		

6.1.10 Прямий контакт з обладнанням

Важливим є взаємодія з електричним обладнанням, яке використовується під час необхідних вимірів. Недотримання правил безпеки може призвести до негативних наслідків, таких як ураження струмом. Приміщення для вимірів можна віднести до приміщень з підвищеною небезпекою через можливість дотику людини до неструмопровідних частин електроустаткування або металевих конструкцій, які мають контакт із землею. Працівник, виконуючий виміри, має ризик отримати ушкодження через недотримання умов безпеки. Наприклад, це може пошкоджена ізоляція чи інші проводи. В такому випадку потерпілому треба бути першу медичну допомогу чи за необхідністю негайно викликати швидку допомогу.

Напруженість роботи визначається як середня тому що виміри не займають великої кількості часу, а також фізичного навантаження. Звісно, тут присутнє інтелектуальне навантаження, а якщо на це відійде велика кількість часу, працівник може втомитися, що загрожує неправильній роботі чи недостатній увазі при роботі з електроустаткуванням під напругою. Тому для відсутності шкідливого впливу у працівника мають бути перерви.

6.1.11 Регламент робочого часу

Конституція України [23] дає кожному працівнику гарантію на працю, що дає змогу заробляти, на яку працівник погоджується згідно договору при прийнятті на роботу, а також яку він вільно обирає. Також кожна людина має право відпочити, цей час встановлюється за допомогою внутрішнього договору, також встановлюються норми тривалості праці, роботи у нічний час, або скороченого часу роботи. Нормована тривалість праці на підприємстві становить не більше 40 годин на тиждень, але організації при укладенні договору можуть домовлятися про тривалість менше 40 годин.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		66

Також згідно Конституції України встановлюється п'ятиденний робочий тиждень, який має 2 вихідних дня. При складанні внутрішнього договору визначається внутрішній трудовий розпорядок, який встановлює графік змін. Його повинен затвердити власник організації або установи.

Щодо тривалості щоденної зміни, її час також визначається внутрішнім розпорядком або графіками змін. Неповний робочий день або тиждень також має бути обумовлений між працівником та власником підприємства при прийнятті на роботу чи вже в процесі.

Якщо підприємство має нічний зміни, має бути також встановлена нормована тривалість роботи, але на відміну від зміни вдень, нічна скорочується на одну годину. Проте, якщо поперед цього був договір про скорочену роботу праці, це правило не поширюється.

6.1.12 Заходи безпеки при обслуговуванні акумуляторів

Згідно правилам безпеки [24] акумуляторні батареї потрібно тримати в чистоті, раз на 15 днів протирати поверхню, для цього можна змочити ганчірку в 10%-му складі нашатирного спирту. Це потрібно для чистки сірчаної кислоти, потім протерти акумулятор ще раз сухою ганчіркою. Профілактикою також буде систематичне чищення отворів в пробках, це запобігає створенню надлишкових газів. Затискачі потрібно чистити за допомогою технічного вазеліну. Це запобігає їх окисненню.

Щоб зняти чи перенести акумулятор, потрібно також вживати спеціальних заходів. По-перше, слід мати спеціальний одяг, гумовий фартух, рукавички та чоботи. Це запобігає виклику опіків на шкірі, так як електроліт серйозно діє на звичайні одяг та взуття. При потраплянні рідини на шкіру, треба змити її розчином нашатирного спирту у воді.

					7.141.170033.ПЗ	Арк.
						67
Изм.	Лист	№ докум.	Підпис	Дата		

В приміщені, пов'язаному з обслуговуванням акумуляторних батарей забороняється зберігати їжу чи палити. Також забороняється використовувати освітлення при роботі з використанням сірників, свічок тощо. Треба користуватися тільки переносною лампою, наприклад, на 6 В або 12 В. Це пов'язано з тим, що при обслуговуванні акумуляторів виділяється гримучий газ, який дуже легко запалюється. Це може призвести до вибуху в приміщені. Також через це не можна користуватися вилкою навантаження під час зарядки батареї. Під час такого процесу не слід нахилитися до акумулятора, бо при зарядці виходять гази з вентиляційних отворів пробок, що може викликати пошкодження дихальних шляхів чи слизової очей.

6.2 Дії працівників в надзвичайних ситуаціях

Під час вимірювальної роботи з електроприладами, як і при іншій праці з електроустаткуванням під напругою можливі надзвичайні ситуації. В цьому підрозділі буде розглянуто електробезпеку та пожежну безпеку.

6.2.1 Ураження струмом

Згідно інструкції [25] представимо дії першої допомоги потерпілому при ураженні електричним струмом:

1. Потрібно звільнити постраждалого від дії струму. Це найголовніший пункт допомоги, адже ця швидка дія може допомогти зберегти здоров'я або навіть життя працівнику.
- Вимкнути мережу живлення та струмоведучі частини.
 - Відкинути провід від працівника, ураженого струмом, яких знаходиться під напругою, відтягнути його від проводів установки.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		68

В нашому випадку напруга не буде перевищувати 1000В, тому можна буде взяти потерпілого за сухий одяг, при цьому не торкаючись відкритих частин тіла.

Для безпеки можна встати на ізольований предмет, а також використати гумові рукавички.

2. Покласти потерплого спиною на тверду поверхню.
3. Перевірити, чи є у працівника дихання, а також пульс на сонній артерії.
4. Перевірити стан зіниці тому, що широка вказує на погіршення кровопостачання.
5. У будь-якому випадку треба викликати лікаря, незалежно від того, який стан має потерпілий.
6. Якщо постраждалий знаходиться при свідомості, треба покласти його у зручне положення до прибуття швидкої допомоги. Головне, спостерігати за пульсом потерпілого та забезпечити працівнику спокій, не треба рухатись чи продовжувати працю. Якщо виклик швидкої неможливий чи це займе велику кількість часу, краще самостійно доставити постраждалого до медичного пункту.
1. Якщо потерпілий знепритомнів, йому треба розстебнути одяг, перед цим поклавши, забезпечити свіже повітря, спробувати дати нашатирний спирт або бризнути на нього водою та забезпечити спокій. В той же момент потрібно терміново викликати лікаря. Якщо під час очікування у постраждалого погане дихання, необхідно зробити штучне дихання і непрямий масаж серця.

					7.141.170033.ПЗ	Арк.
						69
Изм.	Лист	№ докум.	Підпис	Дата		

7. Якщо не зробити всі зазначені дії вчасно під час відсутності дихання, такі як штучне дихання і зовнішній масаж серця, може настати смерть. Є кілька способів для оживлення організму, які базуються на штучному диханні. Починати його необхідно після звільнення потерпілого від дії електричного струму. Проводити штучне дихання треба до появи позитивного результату або до прибуття лікаря.

Переносити постраждалого не дозволяється, тільки у випадку, коли йому або людині, яка надає першу допомогу може погрожувати небезпека ураження електричним струмом.

6.2.2 Пожежна безпека

У кабінеті для виконання практичних робіт з вимірювань є електричне обладнання, яке може з якихось обставин, наприклад, несправного стану приладу, може призвести до пожежі в робочому приміщенні. Це може також статися через коротке замикання чи неправильну роботу з приладом.

Категорія приміщення за вибухопожежною і пожежною небезпекою, можна надати категорію Г [26]. У робочому приміщенні має бути присутній вогнегасники для тушіння пожежі.

Інструкція згідно ДБН В.1.2-7:2021 [27] для працівників під час виникнення пожежі:

1. За можливості відключити вимірювальний пристрій, осцилограф.
2. Перевірити, чи була викликана пожежна охорона.
3. За можливості, негайно допомогти видалити за межі пожежі всіх працівників, які не пов'язані до тушіння пожежі.

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		70

4. Якщо пожежа виникла на підприємстві, потрібно повідомити про подію керівника установи чи відповідну особу.

Інструкція першої медичної допомоги для працівників, якщо хтось отримав опіки від пожежі (більшість уражень при пожежах – задихаються від продуктів горіння):

1. Якщо у потерпілого горить одяг, його одразу слід скинути чи спробувати погасити вогонь, накривши полум'я ковдрою. Задля недопускання травм шкіри слід відрізати обпалені ділянки одягу.
2. Якщо у постраждалого ознаки першого рівня опіку, слід промити уражені ділянки шкіри антисептиком, після цього обробити за допомогою спирту чи одеколону, а потім накрити чистою тканиною.
3. Якщо у потерпілого є ознаки охолодження, слід зігріти людину, накривши або давши води.
4. При знепритопленні людини можна дати постраждалому вату з нашатирем.
5. Якщо постраждалий не має ознаки дихання, треба почати серцево-легеневу реанімацію.
6. Під час такої реанімації треба негайно викликати бригаду швидкої допомоги.
7. Для знеболення постраждалого треба дати таблетку анальгіну.

ВИСНОВКИ І ЗАГАЛЬНІ РЕКОМЕНДАЦІЇ

Під час виконання кваліфікаційної роботи було проведене теоретичне дослідження з розробки апаратно-програмного комплексу діагностування режимів зарядження тягового акумулятора електромобіля. За допомогою однопалатного мікрокомп'ютеру Arduino було розроблено базову платформу цього комплексу.

Крім того, було проаналізовано стан та роль електромобіля у сьогоденні. Проаналізовані типи існуючих зарядок, їх класифікацію, а також потужність. Ми детально обґрунтували також ті інструменти, які можуть допомогти при розробці системи зарядження. З'ясували, що таке мікрокомп'ютер та серія ESP 32, їх переваги, недоліки та потенціал.

Багато часу було переділено надано на програмний код мікрокомп'ютеру за допомогою середовища Arduino IDE. За допомогою рисунків було представлено основні моменти коду, який бере участь як у програмній частини, так і при розробці самого інтерфейсу.

Результатом цієї роботи є той висновок, що за допомогою спеціальних датчиків струму, напруги, а також певних систем контролю можна розробити власний комплекс для зарядження електромобіля.

Лишається тільки розробити два канали для відображення роботи комплексу зарядки. По-перше, це дисплей, який показує основні параметри, а також режими зарядження, а по-друге – відображення інформації бездротовим способом за допомогою Wi-fi на пристрій с ОС Android, смартфон чи планшет. За допомогою такого пристрою можна також дистанційно керувати пристроєм

Таким чином, це дозволяє власнику електромобіля у домашніх умовах проводити певну діагностику режимів зарядження на відстані, за допомогою телефона.

					7.141.170033.ПЗ	Арк.
						72
Изм.	Лист	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Загальна інформація про електромобільні системи зарядження [Електрон. ресурс] Режим доступу: <https://autogeek.com.ua/cho-nuzhno-znat-o-tipah-zaryadok-dlya-elektromobiley/>;
2. Класифікація заряджувальних систем електроавтомобіля [Електрон. ресурс] Режим доступу: <https://autogeek.com.ua/tipyi-zaryadok-elektromobiley-kakoy-vyibrat-chtobyi-zaryazhatsya-v-ukraine/>;
3. Прикладки моделей зарядок різних типів [Електрон. ресурс] Режим доступу: <https://ekozaryad.com/ua/page/p-charge>;
4. Мікроконтролер Arduino [Електрон. ресурс] Режим доступу: <https://bitkit.com.ua/shho-take-arduino>;
5. Склад мікрокомп'ютера [Електрон. ресурс] Режим доступу: <https://uk.wikipedia.org/wiki/Arduino>;
6. Документація про мікроконтролер Arduino diecimila [Електрон. ресурс] Режим доступу: <https://docs.arduino.cc/retired/boards/arduino-diecimila>;
7. Документація про мікроконтролер Arduino pro mini [Електрон. ресурс] Режим доступу: <https://docs.arduino.cc/retired/boards/arduino-pro-mini>;
8. Документація про мікроконтролер Arduino Fio [Електрон. ресурс] Режим доступу: <https://docs.arduino.cc/retired/getting-started-guides/Arduino-Fio>;
9. Документація про мікроконтролер Arduino Mega Adk Rev3 [Електрон. ресурс] Режим доступу: <https://docs.arduino.cc/retired/boards/arduino-mega-adk-rev3>;
10. Документація про мікроконтролер Arduino main board [Електрон. ресурс] Режим доступу: <https://docs.arduino.cc/retired/boards/lilypad-arduino-main-board>;
11. Переваги ESP-32 [Електрон. ресурс] Режим доступу: https://ela.kpi.ua/bitstream/123456789/35113/1/Khyzhniak_bakalavr.pdf;

					7.141.170033.ПЗ	Арк.
Изм.	Лист	№ докум.	Підпис	Дата		73

12. Характеристики ESP-32 [Електрон. ресурс] Режим доступу: https://habr.com/ru/company/epam_systems/blog/522730/;
13. Офіційний сайт для встановлення Arduino IDE [Електрон. ресурс] Режим доступу: <https://www.arduino.cc/en/software>;
14. Форум для програмування мікроконтролера [Електрон. ресурс] Режим доступу: <https://forum.arduino.cc/t/evse-charger-miev-leaf-tesla-6-16a-esp32/878422>;
15. ПУЕ. Правила улаштування електроустановок [Електрон. ресурс] Режим доступу: <https://art-energetyka.com.ua/Правила-улаштування-електроустановок.pdf>;
16. ДСН 3.3.6.042-99. Санітарні норми мікроклімату виробничих приміщень [Текст]/ Постанова № 42 від 01.12.99 - Київ
17. Дослідження запиленості повітря робочої зони [Електрон. ресурс] Режим доступу: <https://studfile.net/preview/9190993/>;
18. НПАОП 0.00-7.15-18. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями [Текст]/ наказ від 14.02.2018 № 207;
19. ДСН 239-96. Про затвердження державних санітарних правил та норм [Текст]/ наказ від 01.08.1996 № 239;
20. Нормування та захист від електромагнітних випромінювань [Електрон. ресурс] Режим доступу: <https://cpo.stu.cn.ua/Oksana/posibnik/910.html>;
21. ДСН 3.3.6.037-99. Санітарні норми виробничого шуму, ультразвуку та інфразвуку [Текст]/ Постанова від 01.12.1999 № 37;
22. ДСанПіН 3.3.2-007-98. Державні санітарні правила і норми [текст]/ постанова від 10.12.1998;
23. Робочий час та його тривалість [Електрон. ресурс] Режим доступу: https://minjust.gov.ua/m/str_8396;

24. Техніка безпеки при обслуговуванні акумуляторів [Електрон. ресурс]
Режим доступу:
<https://budtehnika.pp.ua/7983-pravila-tehnki-bezpeki-pri-robot-z-akumulyatormimi-batareyami.html>;
25. Перша допомога при ураженні електричним струмом [Електрон. ресурс]
Режим доступу: <https://bozhedarivska-selrada.gov.ua/news/1576497483/>;
26. НАПБ А. 01.001-2014. Про затвердження Правил пожежної безпеки в Україні [текст]/ постанова від 30.12.2014 № 1417;
27. ДБН В.1.2-7:2021. Пожежна безпека [Електрон. ресурс] Режим доступу:
https://www.minregion.gov.ua/wp-content/uploads/2022/06/dbn-v.1.2-7_2021.pdf;
28. Розрахунок ШІМ – сигналу [Електрон. ресурс] Режим доступу:
<https://ev-olution.yolasite.com/DIY-Arduino-EVSE.php?fbclid=IwAR23UIyWYvearjFswAKCXX8PWCxq1a4q0NeUoIQT9HDftjr3ByCuptJ1D0w>
29. СТ-сенсори, Ардуїно [Електрон. ресурс] Режим доступу:
<https://learn.openenergymonitor.org/electricity-monitoring/ct-sensors/interface-with-arduino>
30. Вимірювання напруги змінного струму за допомогою адаптера змінного струму [Електрон. ресурс] Режим доступу:
<https://learn.openenergymonitor.org/electricity-monitoring/voltage-sensing/measuring-voltage-with-an-acac-power-adapter>
31. Датчик струму - 30А [Електрон. ресурс] Режим доступу:
<https://www.sparkfun.com/products/11005>

СПИСОК ГРАФІЧНИХ РОБІТ

1. Класифікація апаратно-програмних комплексів.
2. Переваги серії мікроконтролерів ESP-32.
3. Код інтерфейсу програми.
4. Інтерфейс розробленої програми у смартфоні.

					7.141.170033.ПЗ	Арк.
						76
Изм.	Лист	№ докум.	Підпис	Дата		

ДОДАТОК А

					7.141.170033.ПЗ	Арк.
						77
Изм.	Лист	№ докум.	Підпис	Дата		

КЛАСИФІКАЦІЯ АПАРАТНО-ПРОГРАМНИХ КОМПЛЕКСІВ



Рис. 1 Зарядки Type 1 та Type 2



Рис. 2 Зарядки Type 3

1. Type 1, максимальна потужність - 1.92 кВт;
2. Type 2, максимальна потужність - 7 кВт;
3. Type 3, максимальна потужність - 50 кВт;

					7.141.170033.01			
013	Лист.	№ Докум.	Підпис	Дата	Графічна частина	Літ.	Лист	Листів
Розроб.	Василенко О.О.	А.Василенко	01.12.22					
Перевір.	Муха А.М.	А.Муха	11.12.22				78	81
						МОН України, УДУНТ Кафедра ЕТЕМ, група ЕП2121		
Н. Контр.	Карзова О.О.	О.Карзова	11.12.22					
Затв.	Муха А.М.	А.Муха	11.12.22					

ПЕРЕВАГИ СЕРІЇ МІКРОКОНТРОЛЕРІВ ESP-32

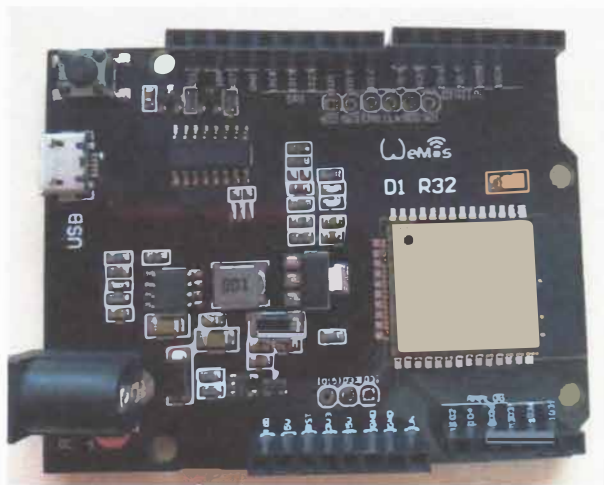


Рис. 3 – Мікрокомп'ютер ESP-32 Wemos D1 Mini

- Низьке енергоспоживання;
- Недорога вартість;
- Вбудований Wi-Fi модуль;
- Простота у користуванні
- Розвинена периферія
- Легка інтеграція



					7.141.170033.02				
013	Лист.	№ Докум.	Підпис	Дата	Графічна частина	Літ.	Лист	Листів	
Розроб.	Василенко О.О.	Р.Василенко	01.12.22						
Перевір.	Муха А.М.	А.Муха	01.12.22				79	81	
Н. Контр.	Карзова О.О.	О.Карзова	01.12.22			МОН України, УДУНТ Кафедра ЕТЕМ, група ЕП2121			
Затв.	Муха А.М.	А.Муха	01.12.22						

КОД ІНТЕРФЕЙСУ ПРОГРАМИ

Графічні частини функцій та налаштувань початкового стану програми

```
typedef enum ChargingState {  
    STATE_A, // Початкова напруга 12V, Not Connected,  
    // ADC: 5050  
    STATE_B, // Початкова напруга 9V, Connected,  
    // ADC: 3400  
    STATE_C, // Початкова напруга 6V, Charging,  
    // ADC: 2847  
    STATE_D, // Початкова напруга 4V, Ventilation F  
    // ADC: 2308  
    STATE_E, // Початкова напруга 0V, No Power  
    // ADC: 1943  
    STATE_F, // Початкова напруга 14V, Full Charge  
    // ADC: 4100  
    STATE_CUSTOM_OFF, // калібрувальний стан  
    // для вимірювання автоматизації,  
    // без статичних клавіш до STATE_CUSTOM_ON or STATE_A  
    STATE_CUSTOM_ON  
};
```

Рис. 4 – Функція початкового стан

```
float ampvalue = 0.00;  
String message = "";  
String amperes1 = "14";  
int minvaluePilot = 0;
```

Рис.5 – Поточні значення датчика

```
const char *ssidArray[] = {  
    "Your_SSID_HERE", "Your_2nd_SSID_HERE" ,  
    "Your_Telephone_Hot_Spot_SSID"  
};
```

Рис. 6 - Wi-fi налаштування

```
const char* ssid = "";  
const char* password = "";
```

Рис. 7 – данні, які заповнюються під час знаходження мережі

					7.141.170033.03				
013	Лист.	№ Докум.	Підпис	Дата	Графічна частина	Літ.	Лист	Листів	
Розроб.	Василенко О.О.	А.Вас	01.12.22						
Перевір.	Муха А.М.	Муха	01.12.22				80	81	
Н. Контр.	Карзова О.О.	О.О.	01.12.22						
Затв.	Муха А.М.	Муха	01.12.22						
						МОН України, УДУНТ Кафедра ЕТЕМ, група ЕП2121			

ІНТЕРФЕЙС РОЗРОБЛЕНОЇ ПРОГРАМИ У СМАРТФОНІ

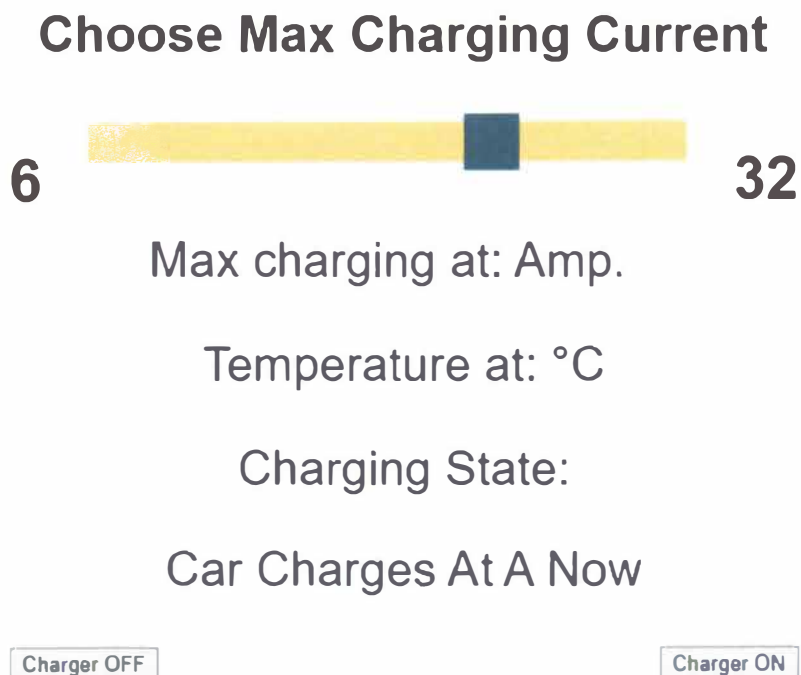






Рис. 8 – Інтерфейс програми

На рис. 8 представлено можливість вибору максимального заряду, при цьому нам доступні температурні показники, стан комплексу заряду та сповіщення про те, чи включена система зарядження в цей момент.

Для відображення режимів роботи комплексу застосовується два канали:

- перший - це дисплей, або індикаторна панель, для відображення основних параметрів та режимів роботи зарядного комплексу;
- другий – це вивід інформації на Андроїд системи, з можливістю дистанційного контролю та керування.

					7.141.170033.04			
013	Лист.	№ Докум.	Підпис	Дата	Графічна частина	Літ.	Лист	Листів
Розроб.	Василенко О.О.		01.12.22				81	81
Перевір.	Муха А.М.		14.12.22					
Н. Контр.	Карзова О.О.		14.12.22					
Затв.	Муха А.М.		14.12.22					
						МОН України, УДУНТ Кафедра ЕТЕМ, група ЕП2121		