

Міністерство освіти і науки України
Український державний університет науки і технологій

Факультет «Управління енергетичними та економічними процесами»

Кафедра «Електротехніка та електромеханіка»

ЗАТВЕРДЖУЮ

Завідувач кафедри ЕТЕМ


(підпис) /Андрій МУХА/

Дата 20.12.23

Пояснювальна записка
до кваліфікаційної роботи магістра

на тему: «Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля»
за освітньою програмою: «Енергетичні та електромеханічні системи на транспорті»
зі спеціальності: «141 Електроенергетика, електротехніка та електромеханіка»

Виконав: студент
групи «ЕЕ2221»


(підпис студента)

/Валентин
ВАСИЛЬКОВСЬКИЙ/
(Ім'я ПРІЗВИЩЕ)

Керівник:


(підпис)

/проф., зав.каф. Андрій
МУХА/
(посада, Ім'я ПРІЗВИЩЕ)

Нормоконтролер:


(підпис)

/доц. Оксана КАРЗОВА/
(посада, Ім'я ПРІЗВИЩЕ)

Консультанти:

(назва розділу)

(підпис)

//

(посада, Ім'я ПРІЗВИЩЕ)

(назва розділу)

(підпис)

//

(посада, Ім'я ПРІЗВИЩЕ)

(назва розділу)

(підпис)

//

(посада, Ім'я ПРІЗВИЩЕ)

(назва розділу)

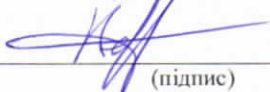
(підпис)

//

(посада, Ім'я ПРІЗВИЩЕ)

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

Студент


(підпис)

Дніпро – 2023 рік

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

Faculty «Management of energy and economic processes»
Department «Electrical engineering and electromechanics»

Explanatory Note
to Master's Thesis

on the topic: «Research of transient processes in the control circuits of the charging complex of an electric vehicle»

according to educational curriculum «Energy and electromechanical systems in transport»

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

Done by the student of the group EE2221:  /Valentyn VASYLKOVSKYI/

Scientific Supervisor:  /Andrii MUKHA/

Normative controller:  /Oksana KARZOVA/

Supervisors

_____	//
(Chapter title heading)	(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 та економічними процесами»

Кафедра: «Електротехніка та електромеханіка»

Рівень вищої освіти: магістр

Освітня програма: «Енергетичні та електромеханічні системи на транспорті»

Спеціальність: «141 Електроенергетика, електротехніка та електромеханіка»

ЗАВДАННЯ

на кваліфікаційну роботу магістра

студенту Васильковський Валентин Андрійович

1. Тема роботи: «Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля»

Керівник роботи: Муха Андрій Миколайович, професор, завідуючий кафедрою

затверджені наказом № 56 ст від 18.01.2023

2. Строк подання студентом роботи: 20.12.2023 р.

3. Вихідні дані до роботи: Теорія перехідних процесів в електричних колах та технічна документація на електромобіль

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

4.1 Аналітична частина: Режим роботи аналогової системи контролю заряду бортового акумулятора

4.2 Основна частина: Розробка алгоритму керування режимами зарядного пристрою по системі Flowcode

4.3 Охорона праці та захист навколишнього середовища: Вимоги до кваліфікаційних груп по електробезпеці та порядок отримання кваліфікаційних груп

4.4 Економічна частина:

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): 1)Розробка принципової блок-схеми. 2)Розробка базової схеми підключення Arduino Uno. 3)Розробка блок-схеми напруги пілот

сигналу та значення струмової установки. 4)Написання коду в Arduino IDE.

5)Завантаження коду на плату Arduino UNO. 6)Заміри напруги на платі.

6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Завдання видав: (підпис консультанта, дата)	Завдання прийняв: (підпис студента, дата)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вступ	01.09.2023	5%
2	Режими роботи аналогової системи контролю заряду бортового акумулятора	14.09.2023	20%
3	Розробка алгоритму керування режимами зарядного пристрою по системі Flowcode	12.10.2023	25%
4	Реалізація програмно керованого комплексу системи на базі комп'ютера Arduino	09.11.2023	25%
5	Вимоги до кваліфікаційних груп по електробезпеці та порядок отримання кваліфікаційних груп	23.11.2023	15%
6	Висновки	06.12.2023	4%
7	Списки використаних джерел	11.12.2023	6%
8	Подання кваліфікаційної роботи до кафедри	20.12.2023	100%
9	Захист кваліфікаційної роботи на засіданні Екзаменаційної комісії	24.01.2024	

Студент


(підпис)

Валентин

ВАСИЛЬКОВСЬКИЙ

(Ім'я ПРІЗВИЩЕ)

Керівник роботи






(підпис)

проф., зав.каф. Андрій

МУХА

(Ім'я ПРІЗВИЩЕ)

№ рядка	Формат	Позначення	Назва	Кільк. арк.	№ екз.	Прим
1			<u>Документація загальна</u>			
2			Заново розроблена			
3	A4	7.141.190034.ПЗ	Пояснювальна записка	71		
4						
5			Запозичена			
6						
7			<u>Графічна частина</u>			
8			Заново розроблена			
9	A4	7.141.190034.01	Розробка принципової	1		
10			блок-схеми			
11	A4	7.141.190034.02	Розробка базової схеми	1		
12			підключення Arduino UNO			
13	A4	7.141.190034.03	Розробка блок-схеми напруги	1		
14			пілот сигналу та значення			
15	A4	7.141.190034.04	Написання коду в Arduino IDE	1		
16						
17	A4	7.141.190034.05	Завантаження коду на плату	1		
18			Arduino UNO			
19	A4	7.141.190034.06	Заміри напруги на платі	1		
20						
21			Запозичена			
22						
23			<u>Електронна частина</u>			
24						

					7.141.190034.ПЗ		
					Відомість кваліфікаційної роботи		
Змн.	Арк.	№ докум.	Підпис	Дата	Літ.	Маса	Масштаб
						1	1:1
Розроб.		Васильковський		20.12			
Перевір.		Муха.А.М.		20.12			
Т. Контр.					Арк.	5	Аркушів 79
Реценз.					МОН України. УДУНТ		
Н. Контр.		Карзова О.О.		20.12	Кафедра «ЕТЕМ»		
Затверд.		Муха А.М.		20.12	7.141.190034.ПЗ		

РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи магістра на тему: «Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля» містить: 72 сторінок основного тексту, 105 рисунків, 2 таблиці, 1 додаток, 10 літературних джерел.

Об'єкт розробки — процеси регулювання струмовими уставками та сигналом пілота.

Мета роботи — розробка програмного комплексу контролю струмів заряду та напруги пілота.

Метод дослідження та апаратура — аналітичний та експериментальний.

Результатом роботи є результати експериментальних досліджень.





КЛЮЧОВІ СЛОВА: МІКРОПРОЦЕСОР, БЛОК-СХЕМА,

КОД, ПІЛОТ СИГНАЛ, ЕЛЕКТРОМОБІЛЬ.

					7.141.190034.ПЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

ЗМІСТ

ВСТУП.....	10
РОЗДІЛ 1 РЕЖИМИ РОБОТИ АНАЛОГОВОЇ СИСТЕМИ КОНТРОЛЮ ЗАРЯДУ БОРТОВОГО АКУМУЛЯТОРА	12
1.1 Опис системи.....	12
1.1.2 Інтерфейс пілотного сигналу керування	13
1.1.3 Робочий цикл J1772	13
1.2 Стан сигналу контрольного пілота.....	15
РОЗДІЛ 2 РОЗРОБКА АЛГОРИТМУ КЕРУВАННЯ РЕЖИМАМИ ЗАРЯДНОГО ПРИСТРОЮ ПО СИСТЕМІ Flowcode	17
2.1 Програма Flowcode.....	17
2.1.2 Меню та панель інструментів	21
2.1.3 Вікно компонентів.....	22
2.1.4 Палітра елементів	37
2.2 Використані компоненти	39
2.2.2 Loop	39
2.2.3 Call Component Macro.....	40
2.2.4 Calculation	42
2.2.5 Delay	43
2.2.6 Call Macro.....	44
2.2.7 Switch.....	45

					Розробка принципової блок-схеми			
					Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля	Літ.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата			1	1 : 1
Розроб.		Васильковський		20.12				
Перевір.		Муха.А.М.		20.12				
Т. Контр.						Арк. 7	Аркушів 79	
Реценз.						МОН України. УДУНТ		
Н. Контр.		Карзова О.О.		20.12	7.141.190034.ПЗ	Кафедра «ЕТЕМ»		
Затверд.		Муха А.М.		20.12				

ВСТУП

Сучасних автомобілістів не дивує збільшення кількості електромобілів на дорогах. Електромобілі мають багато переваг, найголовніша з яких – низька вартість "пального", тобто електроенергії.

Як відомо, вартість електроенергії в нашій країні досить низька, використання електромобілів є вигідним. Тому, незважаючи на те, що вартість покупки електромобіля досить висока в порівнянні зі звичайним транспортним засобом, багато людей вважають за краще купувати саме електромобіль, адже потім це окупиться.

Важливим критерієм при виборі та купівлі електромобіля є вибір зарядного пристрою. У цьому випадку слід бути особливо уважним при виборі правильного варіанту, оскільки від цього механізму залежить час і інтенсивність зарядки.

Існують різні стандарти зарядних пристроїв для електромобілів. Вони поділяються на європейські та американські. Оскільки європейський стандарт вважається більш поширеним, розглянемо його докладніше.

Режим 2 (Mode 2). Дані зарядні пристрої підходять для використання у звичайних розетках і мають дуже повільну зарядку, яка в середньому триває приблизно 24-30 годин. Вони характеризуються наявністю кабелю з високим ступенем захисту від перегріву, що зменшує ймовірність короткого замикання. Вони відносяться до зарядних пристроїв з повільною зарядкою.

Зарядні пристрої для електромобілів можна розділити на два типи: Type 1 і Type 2 відоміші як J1772 і Mennekes. Вони відрізняються роз'ємами, які використовуються для підключення до транспортного засобу і, відповідно, функціональними можливостями [1].

										7.141.190034.ПЗ	Арк.
											10
Змн.	Арк.	№ докум.	Підпис	Дата							

Зарядний пристрій для електромобіля Eveus M16 Light Type1 Schuko, 16 A, 3.7 кВт, 1 фазний, 7 м

Артикул: 143670



Бренд [Eveus](#)
Марка M16 Light Type1
Моделний ряд 2022 рік
Тип [зарядний пристрій для електромобіля](#)
Номинальна напруга 220 В
Номинальна потужність 3700 Вт
Максимальний струм 16 А

8 043 грн.

у наявності

Купити



Більше характеристик +

Зарядний пристрій для електромобіля Eveus M16 Pro Type1 з Wi-Fi, Schuko, 7 - 16 A, 3.7 кВт, 1 фазний, 7 м

Артикул: 143673



Бренд [Eveus](#)
Марка M16 Pro Type1
Моделний ряд 2022 рік
Тип [зарядний пристрій для електромобіля](#)
Дисплей ±
Wi-Fi ±
Номинальна напруга 220 В

11 490 грн.

у наявності

Купити



Більше характеристик +

Рис. 1 – приклади зарядних пристроїв

Type 1 або J1772 – це стандартний 5-контактний роз'єм, який найчастіше використовується на азіатських і американських автомобілях. Його часто використовують для підзарядки від електростанцій і комплексів, які працюють за стандартами режимів 2 і 3. Зарядка здійснюється від однофазної мережі зі змінним струмом, при цьому максимальна напруга буде складати 230В, сила струму може досягати до 32А, а гранична потужність – 7,4кВт.

Type 2 або Mennekes – це контактор з 7-контактним роз'ємом призначений для використання з європейськими електрокарами, а також в деяких адаптованих авто під це зарядний пристрій. Особливістю цього пристрою є можливість використання як однофазної, так і трифазної мережі. При цьому максимальне напруження буде 400В, сила струму досягає до 63А, а потужність – до 43кВт. У більшості випадків робота проводиться з такими характеристиками:

- однофазне підключення – 230В, 32А і 7,4кВт;
- трифазне підключення – 400В, 32А і 22кВт [1].

					7.141.190034.ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 1

РЕЖИМИ РОБОТИ АНАЛОГОВОЇ СИСТЕМИ КОНТРОЛЮ ЗАРЯДУ БОРТОВОГО АКУМУЛЯТОРА

1.1 Опис системи

Електричні транспортні засоби (EVS), включаючи гібридні електромобілі (PHEV), отримують енергію від електричної мережі через обладнання для живлення електромобілів (EVSE), більш відоме як зарядні пристрої для електромобілів. Щоб полегшити подачу електроенергії до автомобіля, EVSE знаходиться між стабільним мережевим з'єднанням і транспортним засобом, як показано на рис. 1.1 [2].

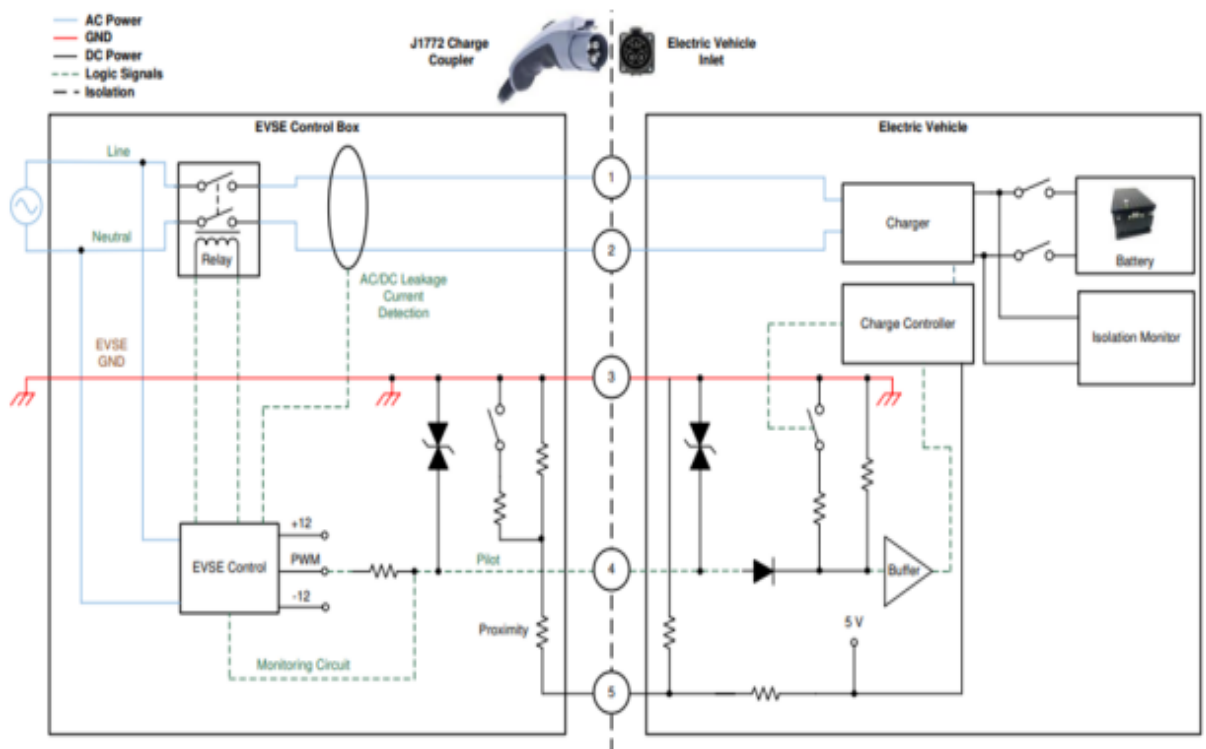


Рис. 1.1 – Конфігурація системи AC рівень 1 і рівень 2

Основні функції EVSE включають:

- регульований електричний струм: гарантує, що оптимальний струм забезпечується і знаходиться в межах максимального струму, який може витримати електрокар;
- виявлення залишкового струму змінного/постійного струму (RCD);
- реле та контакторний привод та виявлення контактів із замком;

Змн.	Арк.	№ докум.	Підпис	Дата

1.2 Стан сигналу контрольного пілота

Підключення та узгодження EVSE відбувається через різні стани ШІМ-сигналу та навантаження опори транспортного засобу. Таблиця 1.2. висвітлює ці стани:

Таблиця 1.2. Параметри стану пілотного сигналу

Стан	Висока напруга пілота	Низька напруга пілота	Частота	Опір	Опис
Стан А	12 В	Незастосовний	Постійний струм	Незастосов ний	Не підключений
Стан В	9 В	-12 В	1 кГц	2,74 кОм	Електромобіль підключено, готовий до заряджання
Стан С	6 В	-12 В	1 кГц	882 Ом	Заряджання EV
Стан D	3 В	-12 В	1 кГц	246 Ом	Заряджання електромобіля, потрібна вентиляція
Стан Е	0 В	0 В	Незастосовний	-	Помилка
Стан F	Незастосовний	-12 В	Незастосовний	-	Невідома помилка

Стани А, В і С є основною функціональністю і визначають нормальну роботу. EVSE зазвичай виконує

кілька самотестувань після першого ввімкнення, а потім переходить у стан А. Коли він готовий, нормальний процес з'єднання

відбувається в кілька етапів:

1. EVSE подає 12 В на пілотний дріт. Ця передача сигналізує автомобілю про те, що штекер підключено.

2. Коли штекер підключено, автомобіль створює навантаження 2,74 кОм на пілотну лінію, що знижує напругу до 9 В.

3. EVSE переходить у стан В, де вмикає ШІМ, який сигналізує автомобілю, скільки струму він може споживати. EVSE також замикає реле, забезпечуючи живлення автомобіля.

4. Автомобіль починає споживати енергію і перемикається на навантаження 822 Ом, що знижує напругу до 6 В, сигналізуючи EVSE про початок заряджання.

5. Більшість транспортних засобів продовжують споживати невелику кількість енергії в стані С, навіть коли вони повністю заряджені, тому процес заряджання закінчується від'єднанням кабелю, що повертає напругу до 12 В. EVSE вимірює цей процес, замикає реле і повертається до стану А.

Додаткова обробка помилок, таких як відсутність діодів у транспортному засобі або неправильне підключення, може бути виявлена і обробляються EVSE також шляхом відключення живлення [2].

					7.141.190034.ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2

РОЗРОБКА АЛГОРИТМУ КЕРУВАННЯМИ РЕЖИМАМИ ЗАРЯДНОГО ПРИСТРОЮ ПО СИСТЕМІ FLOWCODE

2.1 Програма Flowcode

Flowcode - це програма для розробки програмного забезпечення, яка дозволяє створювати електронні пристрої без необхідності писати код. Вона використовує блок-схеми (графічні символи та з'єднувальні лінії), які представляють різні функції та дії пристрою. Можна перетягувати та з'єднувати ці блоки, створюючи програми за допомогою графічного інтерфейсу, що робить його доступним для тих, хто не володіє глибокими знаннями в області програмування [4].

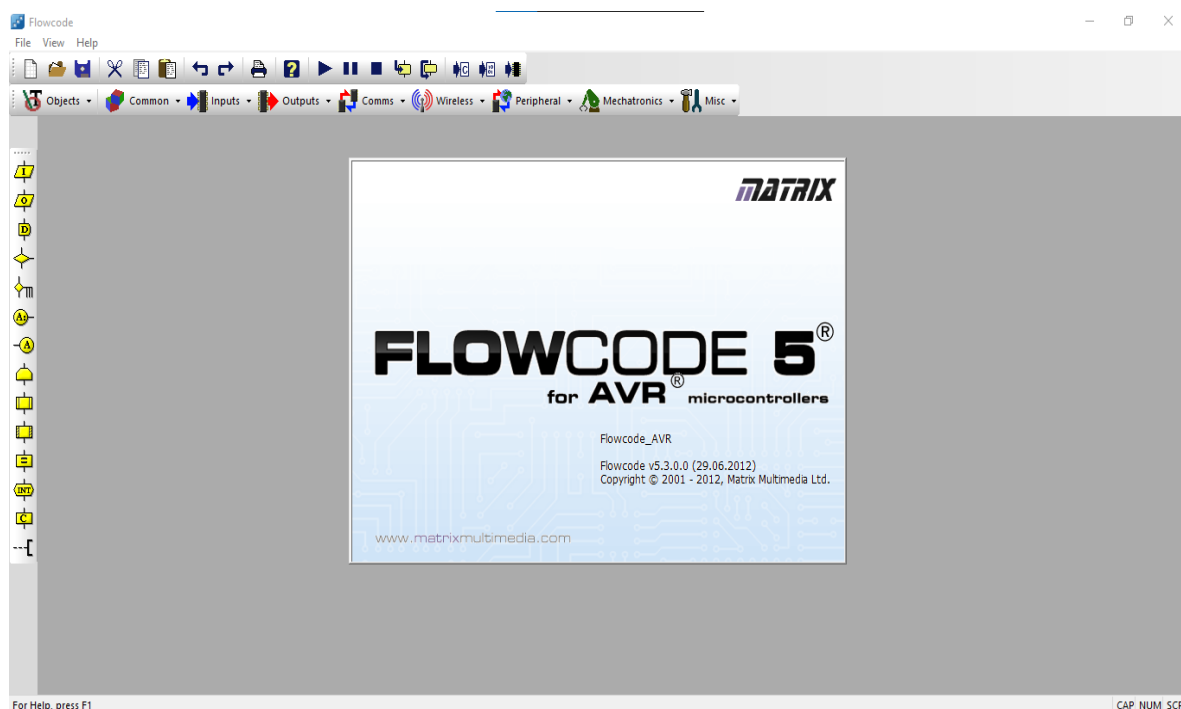


Рис. 2.1 – програмне забезпечення Flowcode

Flowcode розроблений з урахуванням легкості використання, особливо для початківців. Він підтримує багато популярних мікроконтролерів та пристроїв, таких як Arduino, Raspberry Pi, PIC та інші. За допомогою Flowcode можна створювати програми для керування світлодіодами, сенсорами, моторами та багато іншого. Це робить його універсальним для розробки програм для різних вбудованих систем. При створенні нового проекту програма пропонує обрати мікроконтролер з яким ви будете

									Арк.
									17
Змн.	Арк.	№ докум.	Підпис	Дата					

працювати. Як видно на рис. 2.2 вам дається на вибір шість мікроконтролерів, таких як Arduino, ATTINY, ATMEGA, ATXMEGA, AT90, Misc [4].

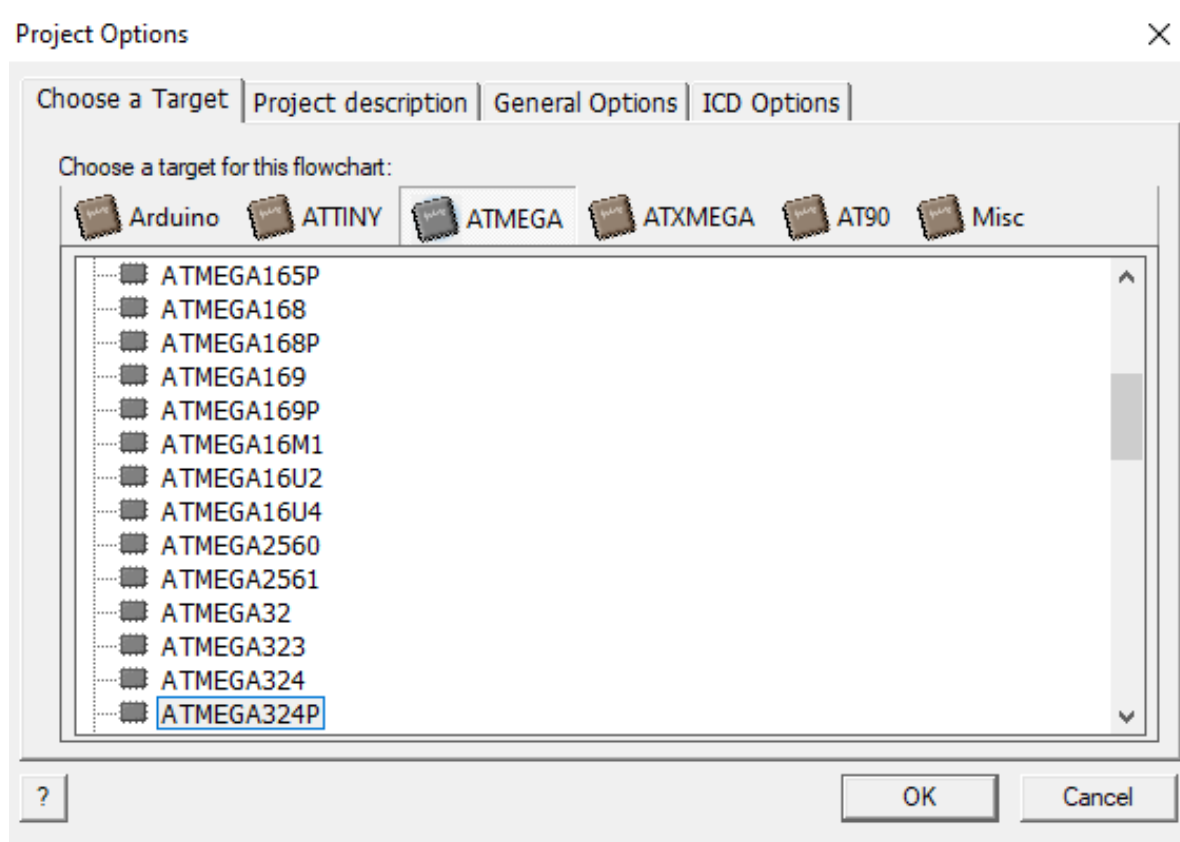


Рис. 2.2 – вибір мікроконтролера

В нашому випадку був обраний мікроконтролер ATMEGA324P, який показаний на рис. 2.2

ATMEGA324P – це 8-бітний мікроконтролер, який входить до сімейства мікроконтролерів AVR, розроблених компанією Atmel. Основні характеристики цього мікроконтролера включають:

1. Архітектура AVR: ATmega324P має архітектуру RISC (Reduced Instruction Set Computing), що сприяє високій продуктивності та ефективності коду.
2. Робоча частота: Здатен працювати на високих частотах, наприклад, до 20 МГц, що дозволяє виконувати завдання з високою швидкістю.
3. Пам'ять: Має вбудовану програмну пам'ять Flash для зберігання програмного коду. Крім того, він має вбудовану EEPROM для зберігання даних і вбудований оперативний запам'ятовуючий пристрій SRAM.

					7.141.190034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

4. Периферійні пристрої: Включає в себе різноманітні периферійні пристрої, такі як таймери/лічильники, порти введення/виведення (GPIO), аналого-цифрові конвертери (ADC), USART (Universal Synchronous and Asynchronous serial Receiver and Transmitter) для зв'язку за стандартом RS232 та інші.

5. Інтерфейси зовнішнього з'єднання: Мікроконтролер має різноманітні інтерфейси для зовнішнього з'єднання, такі як SPI (Serial Peripheral Interface), I2C (Inter-Integrated Circuit), та інші.

6. Низькоспоживана технологія: Підтримує низькоспоживані режими для оптимізації використання енергії та продовження терміну служби вбудованих систем на батареях.

7. Упаковка: АТмега324Р часто доступний в корпусі типу PDIP (Plastic Dual Inline Package), який спрощує використання на експериментальних платах [5].

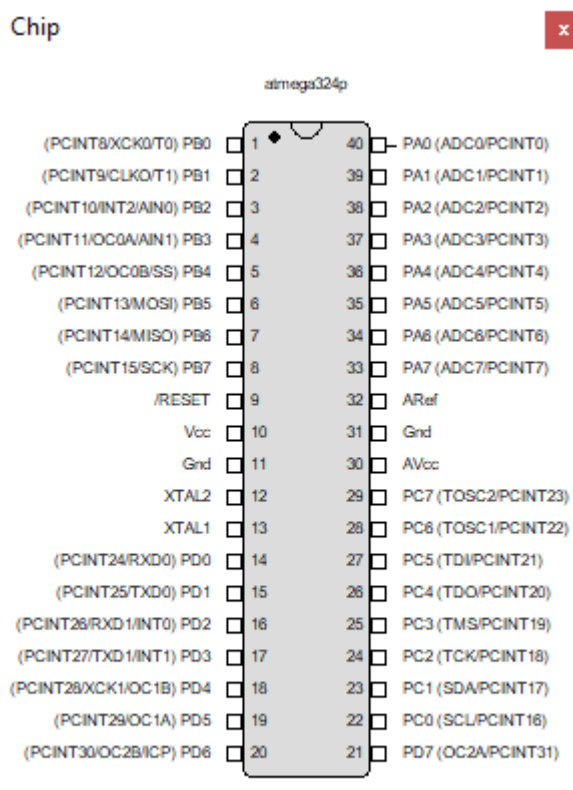


Рис. 2.3 – мікроконтролер АТМЕГА324Р

В новому проєкті також відкривається панель яка вказана на рис. 2.4, на ній будуть розміщені різні елементи які нам будуть необхідні.

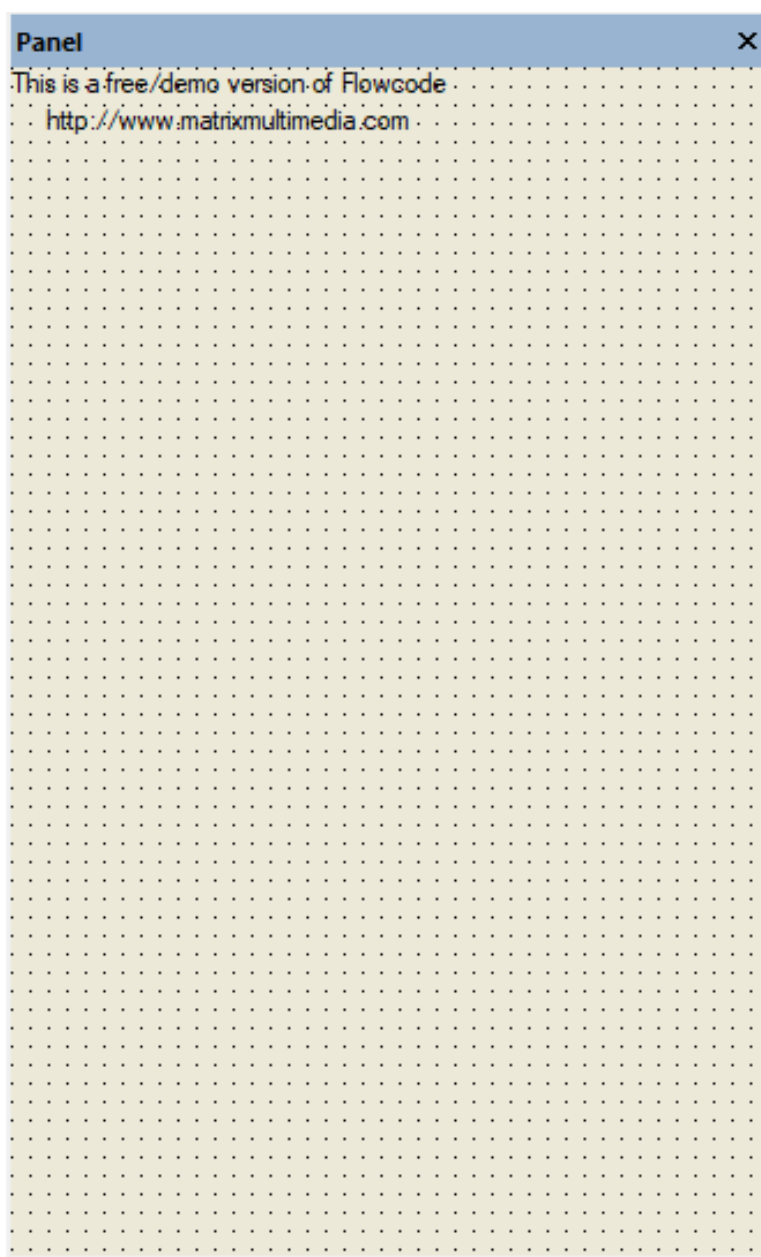


Рис. 2.4 – панель на якій відображаються вибрані елементи

Одним із головних елементів інтерфейсу є робоча область (Workspace): Місце, де користувач створює свою програму. Це може бути представлено у вигляді блок-схеми, де різні блоки відповідають різним операціям чи функціям, вона показана на рис. 2.5, в цій області ми створюємо блок-схеми та перевіряємо її працездатність.

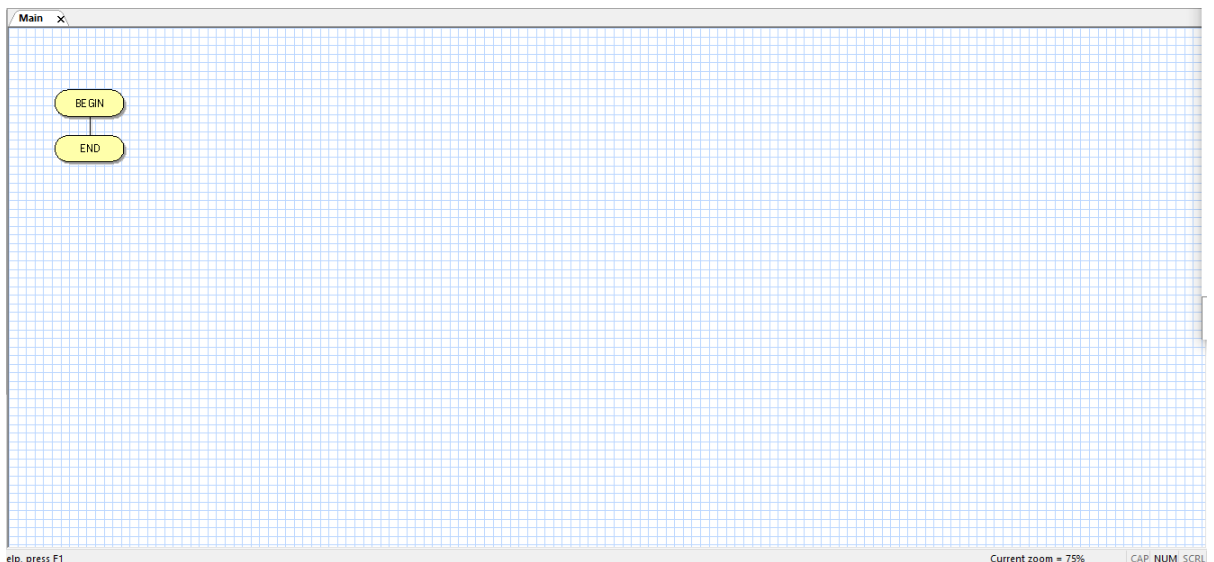


Рис. 2.5 – робоча область (Workspace)

2.1.2 Меню та панель інструментів

Меню та панель інструментів містять різноманітні опції для роботи з проектом, такі як збереження, завантаження, компіляція і завантаження на мікроконтролер показані на рис. 2.6

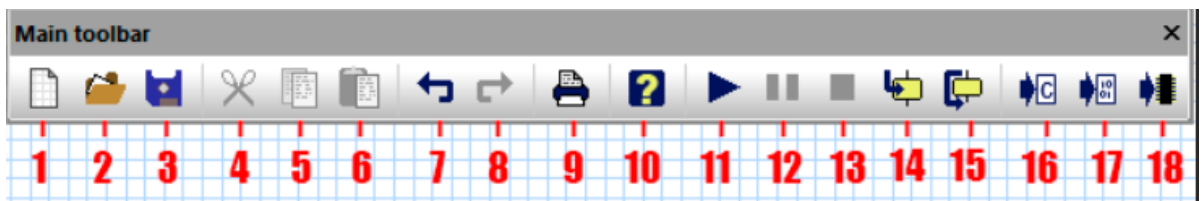


Рис. 2.6 – меню та панель інструментів

В цій панелі вказані наступні інструменти та іконки:

- 1) Створити новий документ
- 2) Відкрити існуючий документ
- 3) Зберегти поточний документ
- 4) Вирізати виділену область та помістити в буфер обміну
- 5) Скопіювати
- 6) Вставити скопійоване
- 7) Повернутись до попередньої дії
- 8) Повернутись до вже виконаної дії
- 9) Роздрукувати поточний документ
- 10) Довідка
- 11) Запустити симуляцію

Змн.	Арк.	№ докум.	Підпис	Дата

- 12) Поставити симуляцію на паузу
- 13) Зупинити симуляцію
- 14) Перейти до блоку
- 15) Переступити через блок
- 16) Скопіювати блок-схему у файл .C
- 17) Скопіювати блок-схему у файл .HEX
- 18) Скопіювати блок-схему та передайте її на мікросхему

2.1.3 Вікно компонентів

Вікно компонентів у Flowcode - це інтерактивний інструмент для вибору і додавання різноманітних компонентів та блоків до вашого проекту. Це важлива частина інтерфейсу Flowcode, яка спрощує розробку програм для вбудованих систем [6].

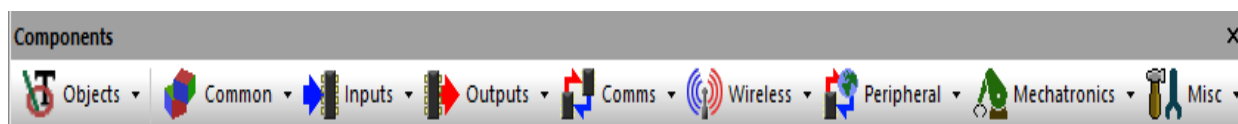


Рис. 2.7 – вікно компонентів

Перший компонент називається «Об'єкти», в ньому є сховані інші компоненти які вказані на рис. 2.8

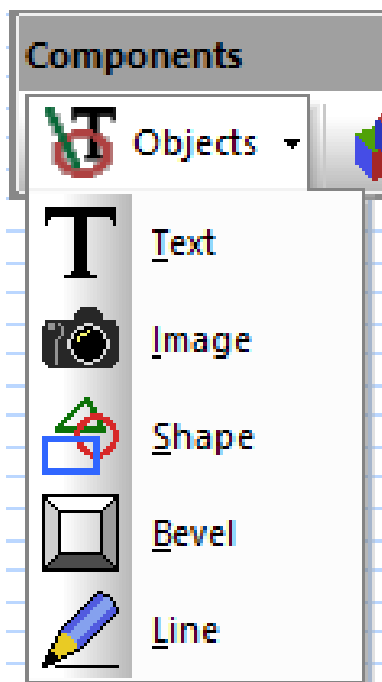


Рис. 2.8 – компонент об'єкти

В ньому можна вставити на робочу область такі компоненти як текст, зображення, форма, фаска та лінія.

Наступний компонент називається «Поширений», в ньому також сховані компоненти, які вказані на рис. 2.9

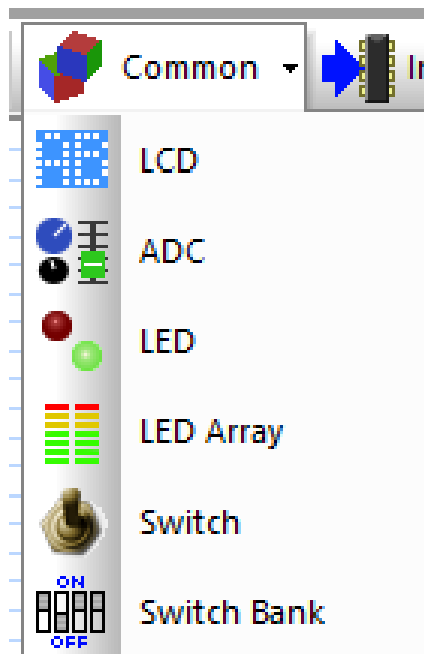


Рис. 2.9 – компонент поширений

В компоненті поширений є різні види компонентів:

1) LCD (Liquid Crystal Display) в Flowcode - це компонент, який може бути використаний для взаємодії з рідкокристалічним дисплеєм (LCD) в мікроконтролерних проектах, вказаний на рис. 2.10

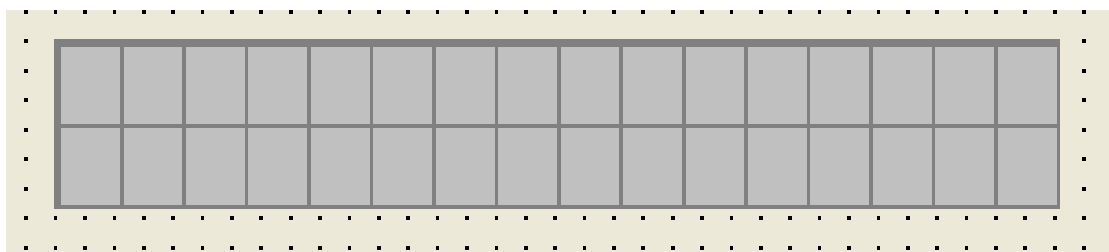


Рис. 2.10 - LCD

2) ADC (Analog-to-Digital Converter) є важливим компонентом у мікроконтролерах і дозволяє зчитувати аналогові сигнали, такі як напруга чи струм, та перетворювати їх у цифровий формат, зрозумілий мікроконтролеру для подальшого оброблення, вказаний на рис. 2.11

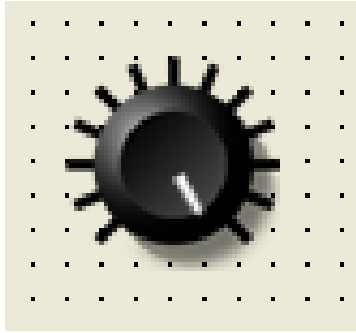


Рис. 2.11 – ADC

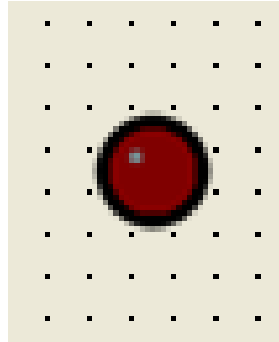


Рис. 2.12 - LED

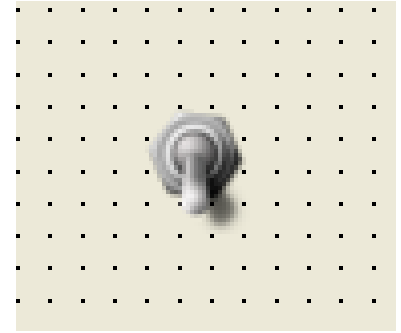


Рис. 2.14 – Switch

3) LED (Light Emitting Diode) це світлодіоди їх можна контролювати за допомогою відповідних компонентів та блоків програмування, вказаний на рис. 2.12

4) LED Array (масив світлодіодів) вказує на групу світлодіодів, які розташовані в певному порядку чи конфігурації на спеціальному підкладі або вбудовані в одну конструкцію, вказаний на рис. 2.13

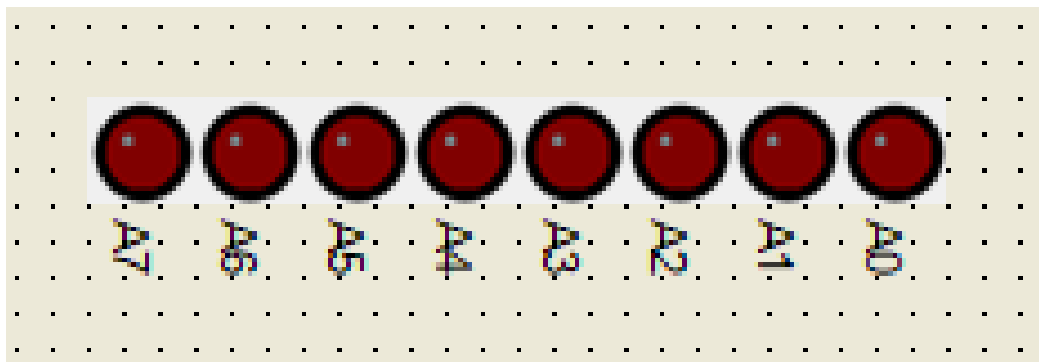


Рис. 2.13 – LED Array

5) Switch (перемикач) використовується для реалізації логіки, пов'язаної з введенням від користувача через фізичний або віртуальний перемикач. Це може бути корисно для взаємодії з користувачем або включення/виключення певних функцій у програмах, вказаний на рис. 2.14

6) Switch Bank зазвичай вказує на групу або банк перемикачів, які можна використовувати для зчитування введених даних від користувача чи управління різними функціями в програмі. Перемикачі в банку можуть бути

розташовані поруч, і вони можуть використовуватися для обробки багатьох введених сигналів одночасно, вказаний на рис. 2.15

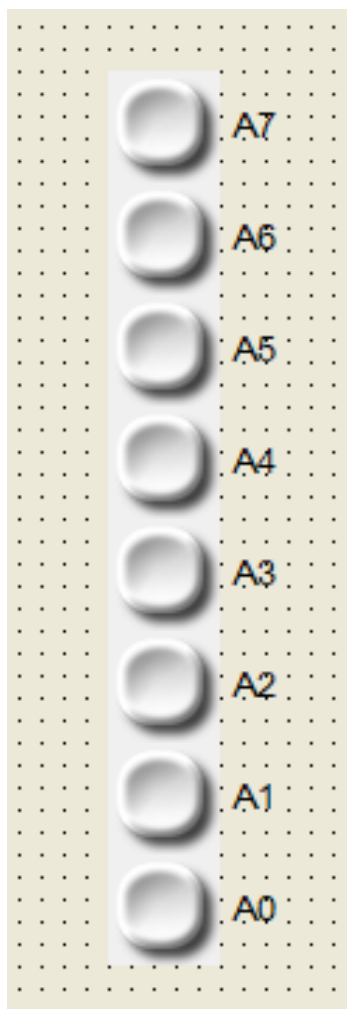


Рис. 2.15 – Switch Bank

Компонент вхід, зображений на рис. 2.16

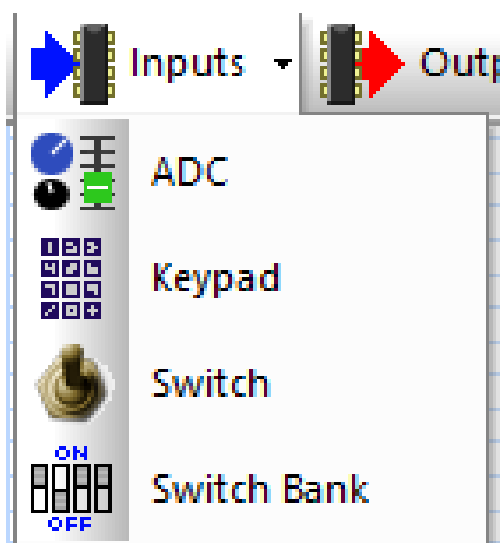


Рис. 2.16 – компонент вхід

Змн.	Арк.	№ докум.	Підпис	Дата

В компоненті вхід, є компонент клавіатура, вказаний на рис. 2.17



Рис. 2.17 - Кеурад

Кеурад (клавіатура) використовується для отримання введених даних від користувача через клавіатуру. Це може бути корисно для взаємодії з користувачем та отримання введених значень для подальшого оброблення в програмі [6].

Компонент вихід зображений на рис. 2.18

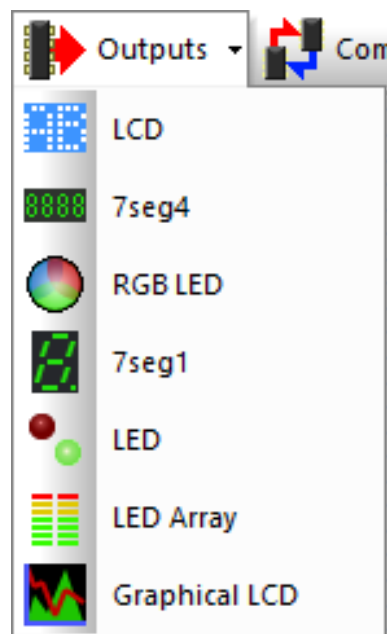


Рис. 2.18 – компонент вихід

В компоненті вихід є різні види компонентів:

1) 7seg4 відноситься до чотирьохзначного семисегментного дисплею, де "7seg" означає семисегментний дисплей, а "4" вказує на кількість цифр на дисплеї. Семисегментний дисплей складається з семи сегментів, які можуть

бути включені або виключені для відображення різних цифр та символів, вказаний на рис. 2.19

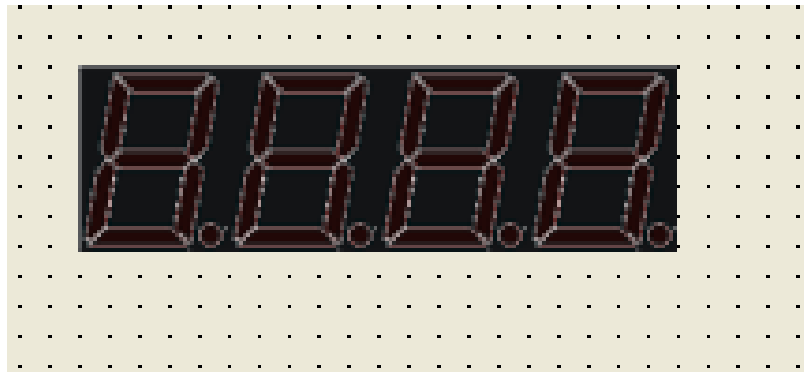


Рис. 2.19 – 7seg4

2) RGB LED - це світлодіод, який може випромінювати світло у трьох основних кольорах: червоному (R - Red), зеленому (G - Green) і синьому (B - Blue). Коли ці тріх кольорів комбінуються в різних пропорціях, можна отримати широкий спектр кольорів, вказаний на рис. 2.20

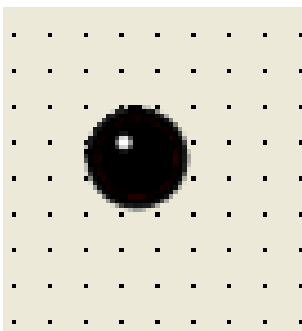


Рис. 2.20 – RGB LED

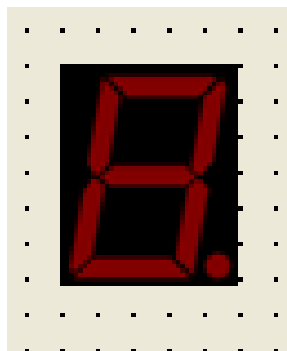


Рис. 2.21 – 7seg1

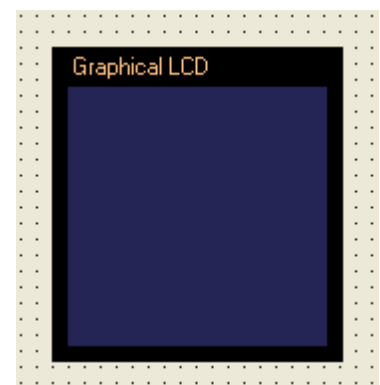


Рис. 2.22 – Graphical LCD

3) 7seg1 - це однозначний семисегментний дисплей. Семисегментний дисплей складається з семи сегментів, які можуть бути включені або виключені для відображення різних цифр та символів, вказаний на рис. 2.21

4) Graphical LCD (Liquid Crystal Display) в Flowcode використовується для відображення зображень, графіки та тексту, вказаний на рис. 2.22

Компонент комунікація, вказаний на рис. 2.23

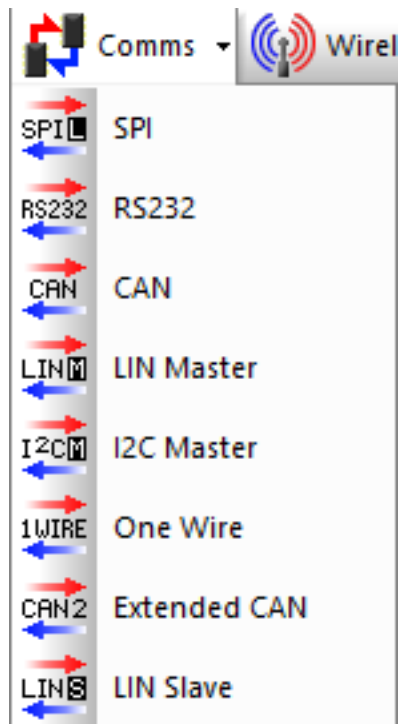


Рис. 2.23 – комунікація

1) SPI (Serial Peripheral Interface) зазвичай включає в себе використання спеціальних компонентів і блоків програмування. SPI - це протокол передачі даних між мікроконтролером та іншими пристроями за допомогою збалансованого набору провідників, вказаний на рис. 2.24

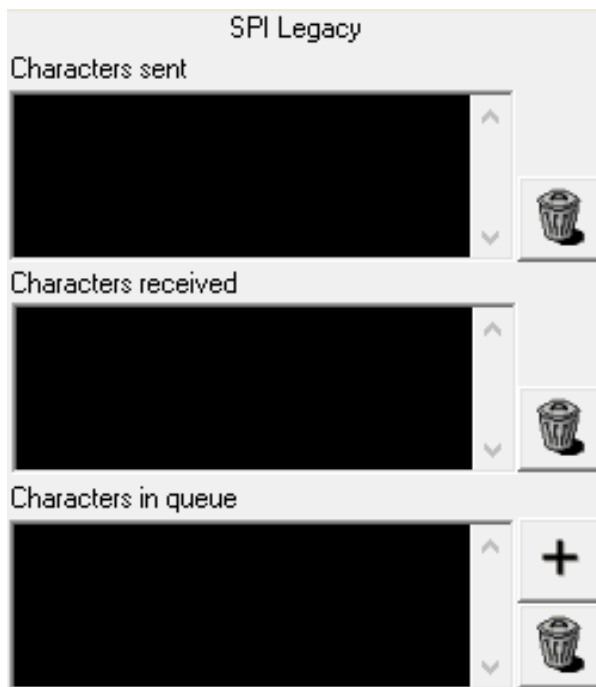


Рис. 2.24 – SPI

2) RS232 - це стандарт для передачі даних між пристроями через серійний інтерфейс, вказаний на рис. 2.25

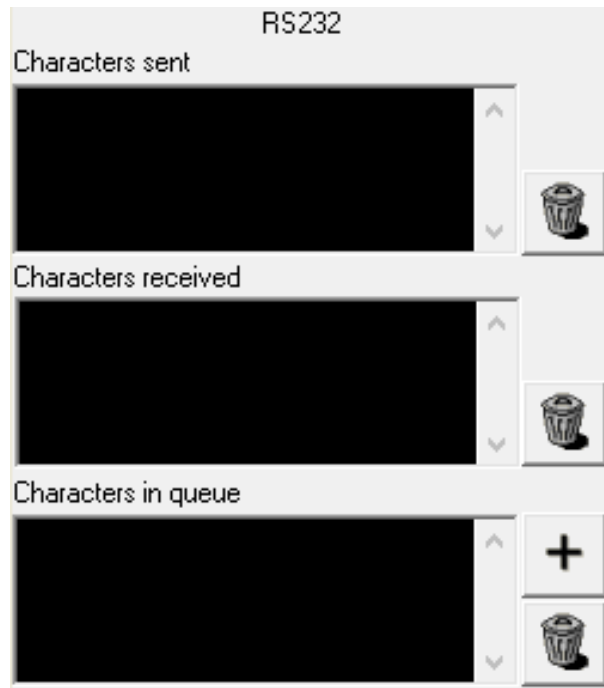


Рис. 2.25 – RS232

3) CAN (Controller Area Network) може включати в себе використання спеціальних компонентів та блоків програмування для роботи з цим протоколом. CAN використовується для обміну даними між пристроями в реальному часі та часто використовується в автомобільній техніці, промислового устаткуванні та інших сценаріях, де важлива стійкість та швидкість передачі даних, вказаний на рис. 2.26

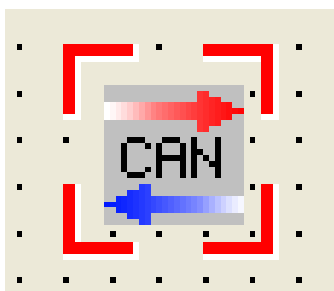


Рис. 2.26 – CAN

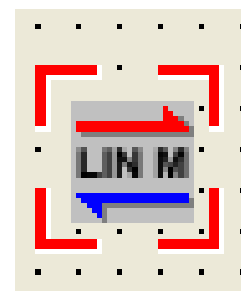


Рис. 2.27 – LIN Master

4) LIN Master підпрограми низького рівня для керування провідним LIN, вказана на рис. 2.27

Бездротові компоненти, вказані на рис. 2.32

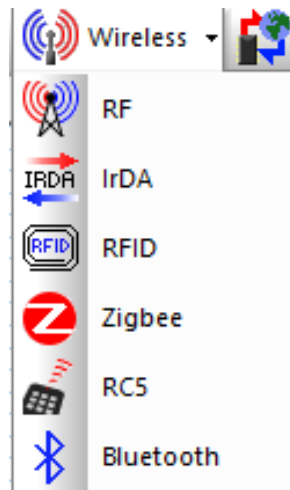


Рис. 2.32 – бездротові компоненти

1) RF – це простий комунікаційний модуль RF-ISM на основі пристрою Microchip Alpha MRF49XA, вказаний на рис. 2.33

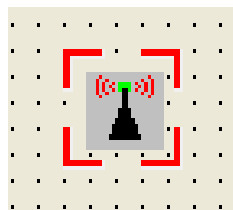


Рис. 2.33 – RF

2) IrDA – це процедури низького рівня для керування стандартним інтерфейсом IrDA, вказані на рис. 2.34

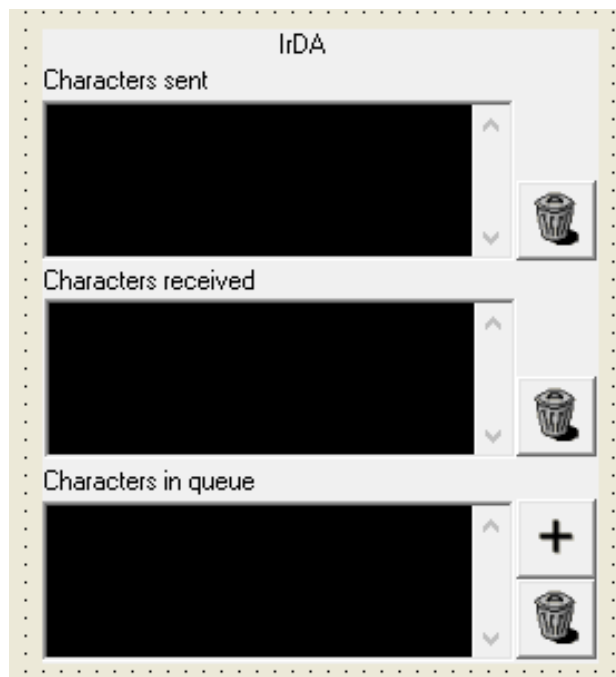


Рис. 2.34 – IrDA

3) RFID – це процедури низького рівня для керування інтерфейсом RFID RF Solutions, вказані на рис. 2.35

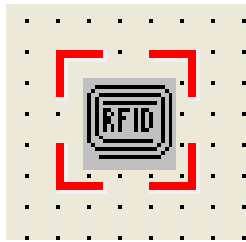


Рис. 2.35 – RFID

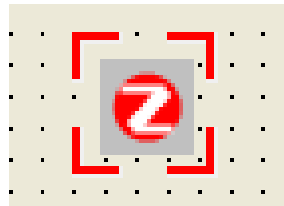


Рис. 2.36 – Zigbee

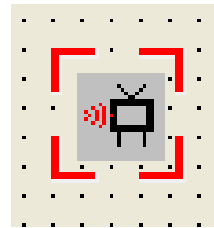


Рис. 2.37 – RC5

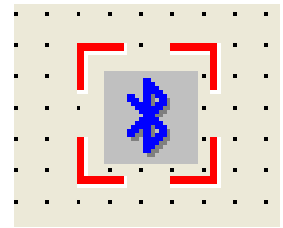


Рис. 2.38 – Bluetooth

4) Zigbee – призначений для роботи з діапазоном модулів XBEE від Digi, вказаний на рис. 2.36

5) RC5 - процедури низького рівня для керування стандартним ПЧ-інтерфейсом RC5, показаний на рис. 2.37

6) Bluetooth – цей модуль дозволяє взаємодіяти з Bluetooth-пристроями, такими як Bluetooth-модулі, смартфони або інші пристрої з підтримкою Bluetooth, вказаний на рис. 2.38 [6]

Периферійні компоненти вказані на рис. 2.39

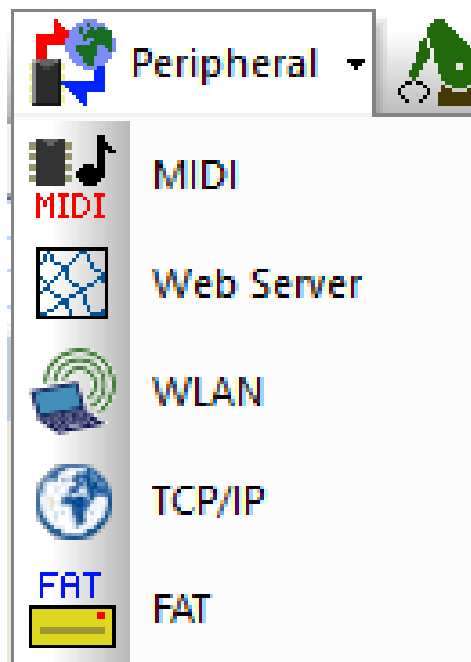


Рис. 2.39 – периферійні компоненти

1) MIDI – це процедури низького рівня для керування або взаємодії зі стандартним MIDI-інтерфейсом, вказаний на рис. 2.40

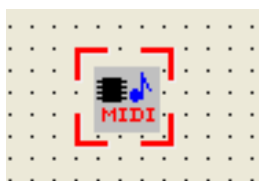


Рис. 2.40 – MIDI

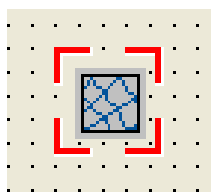


Рис. 2.41 – Web Server

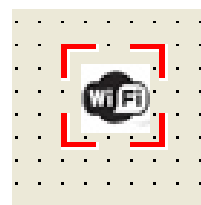


Рис. 2.42 – WLAN

2) Web Server - компонент веб-сервера, призначений для роботи з модулем NM7010A від Wiznet, вказаний на рис. 2.41

3) WLAN – це компонент бездротової локальної мережі, призначений для роботи з модулем ESP8266, вказаний на рис. 2.42

4) TCP/IP – це набір процедур, які використовуються для керування модулем TCP/IP ENC28J60, вказаний на рис. 2.43

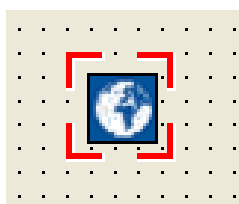


Рис. 2.43 – TCP/IP

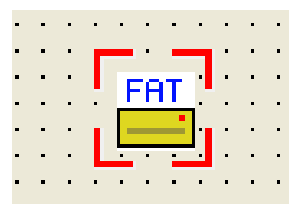


Рис. 2.44 – FAT

5)FAT – компонент, що дозволяє читати та записувати файли на картці SD, SDHC або MMC, вказаний на рис. 2.44 [6]

Компоненти з механотроніки, на рис. 2.45

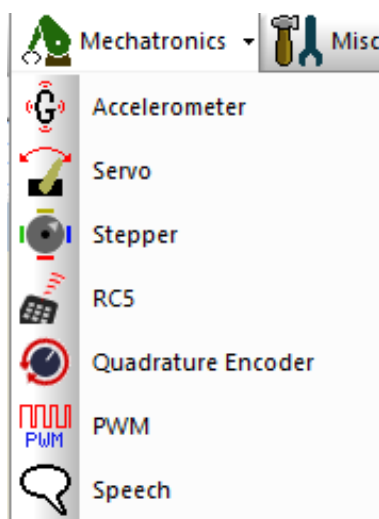


Рис. 2.45 – механотроніка

1) Акселерометр – це LIS3LV02DQ датчик акселерометра з опціями зв'язку I2C або SPI, вказаний на рис. 2.46

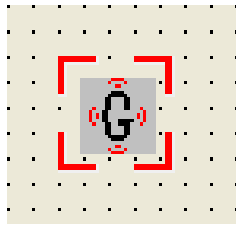


Рис. 2.46 – Акселерометр

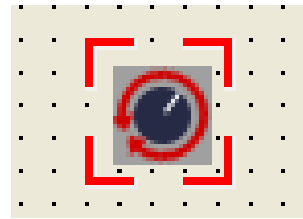


Рис. 2.49 - Quadrature Encoder

2) Сервоконтролер - може керувати до восьми стандартних сервоприводів з ШІМ, вказаний на рис. 2.47

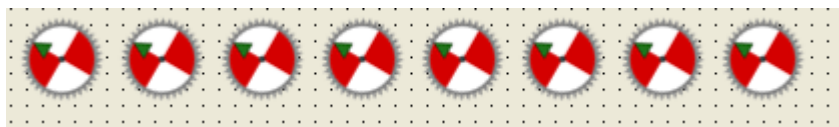


Рис. 2.47 – сервоконтролер

3) Stepper – дозволяє створити компонент крокового двигуна, вказавши об'єкти для його зовнішнього вигляду, вказаний на рис. 2.48

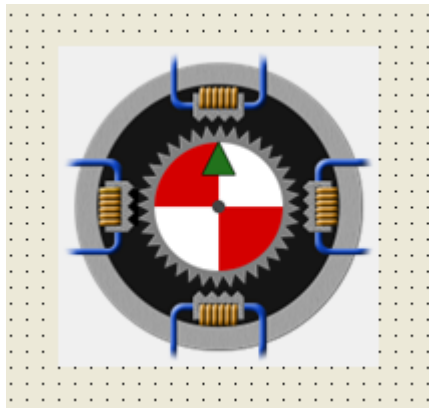


Рис. 2.48 – Stepper

4) Quadrature Encoder - це пристрій, який генерує два сигнали (А та В), які використовуються для вимірювання положення або руху, пристрій вказаний на рис. 2.49

5) PWM – це широтно-імпульсна модуляція, універсальний спосіб генерації цифрового імпульсу за допомогою модуляції позначок/пробілів, вказаний на рис. 2.50 [6]

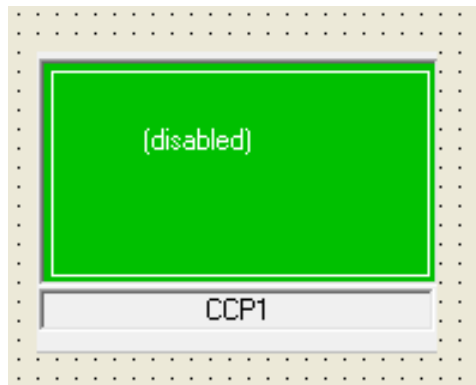


Рис. 2.50 – PWM

б)Speech – це компонент мовлення, що дозволяє використовувати мовлення на основі фонем на вбудованій платформі, вказаний на рис. 2.51



Рис. 2.51 – Speech

Розділ компонентів «Різне», на рис. 2.52

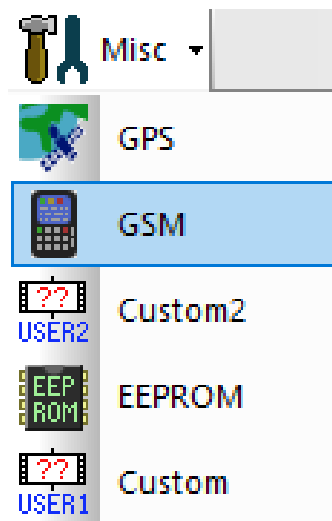


Рис. 2.52 – Різне

1) GPS – це компонент призначений для обробки вхідних даних GPS у стилі NMEA та перетворення їх у значущі значення, такі як довгота, широта, швидкість руху, дата й час за всесвітнім координованим часом. Також доступний у формі електронного блоку GPS EB056, вказаний на рис. 2.53

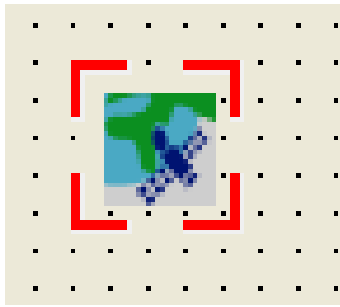


Рис. 2.53 – GPS

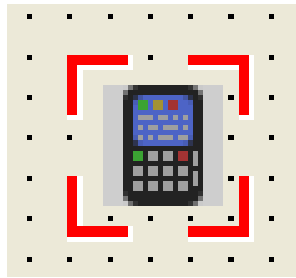


Рис. 2.54 – GSM

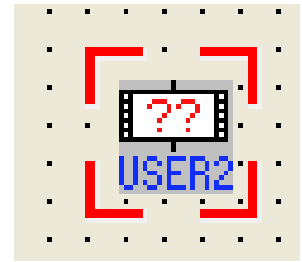


Рис. 2.55 – Custom2

2) GSM – це процедури низького рівня для керування стандартним інтерфейсом AT GSM / GPRS, вказані на рис. 2.54

3) Custom2 – це базовий компонент для створення симуляції серводвигуна, вказаний на рис. 2.55

4)EEPROM – це засіб читання даних із пам'яті EEPROM на борту мікроконтролера та запису даних у пам'ять, показаний на рис. 2.56

Location	Value
0	48
1	27
2	226
3	71
4	24
5	234
6	47
7	97

EEPROM size = 1024 bytes

Рис. 2.56 – EEPROM

5)Custom – це базовий компонент для створення симуляції серводвигуна, вказаний на рис. 2.57

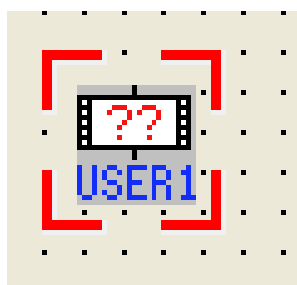


Рис. 2.57 – Custom

2.1.4 Палітра елементів

Звідси користувач може витягти різні блоки або елементи і додавати їх до робочої області. Це можуть бути блоки умов, цикли, операції введення/виведення, таймери та інші. Палітра елементів показана на рис. 2.58

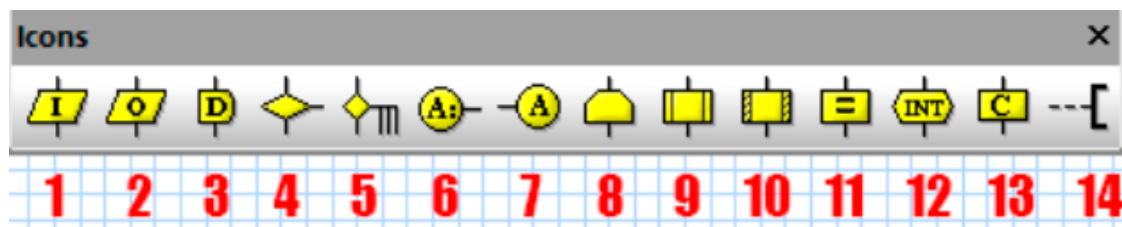


Рис. 2.58 – палітра елементів

В цій палітрі вказані наступні елементи:

- 1) Input – зчитує дані з будь якої ніжки мікроконтролера чи порта в цілому.
- 2) Output – це вивід якогось значення на який порт контролера.
- 3) Delay – це затримка чи пауза.
- 4) Decision – це блок який виконує певні умови.
- 5) Switch – це блок який виконує певну кількість умов за один раз.
- 6) Connection Point – точка з якої виходить програма.
- 7) Goto Connection Point – точка в котру входить програма.
- 8) Loop – це цикл в якому виконуються деякі дії.
- 9) Call Macro – підпрограма, в який виконуються задані дії.
- 10) Call Component Macro – в ньому виконуються інструменти, які встановлені на робочій палені.
- 11) Calculation – виконую різні арифметичні калькуляції.
- 12) Interrupt – виконує переривання в створеній підпрограмі.
- 13) C Code – в цей блок можна вставити код C.
- 14) Comment – можливість залишити коментар в блок-схемі.

Також Flowcode дозволяє симулювати роботу програми без необхідності завантаження коду на реальний пристрій. Це допомагає виявити та усунути помилки до фізичного впровадження програми.

Після того, як створена програма у Flowcode буде завершена, його можна перетворити на виконуваний файл або завантажити безпосередньо на плату мікроконтролера.

У загальному розумінні, Flowcode - це інструмент, який допомагає створювати програми для електронних пристроїв за допомогою графічного інтерфейсу, замість того, щоб писати код вручну. Flowcode робить програмування вбудованих систем доступним та простим для широкого кола користувачів, включаючи ентузіастів, студентів і фахівців у різних галузях [4].

Була побудована блок-схема та підпрограми для неї які показана на рис. 2.59, рис. 2.60 та рис. 2.61 відповідно.

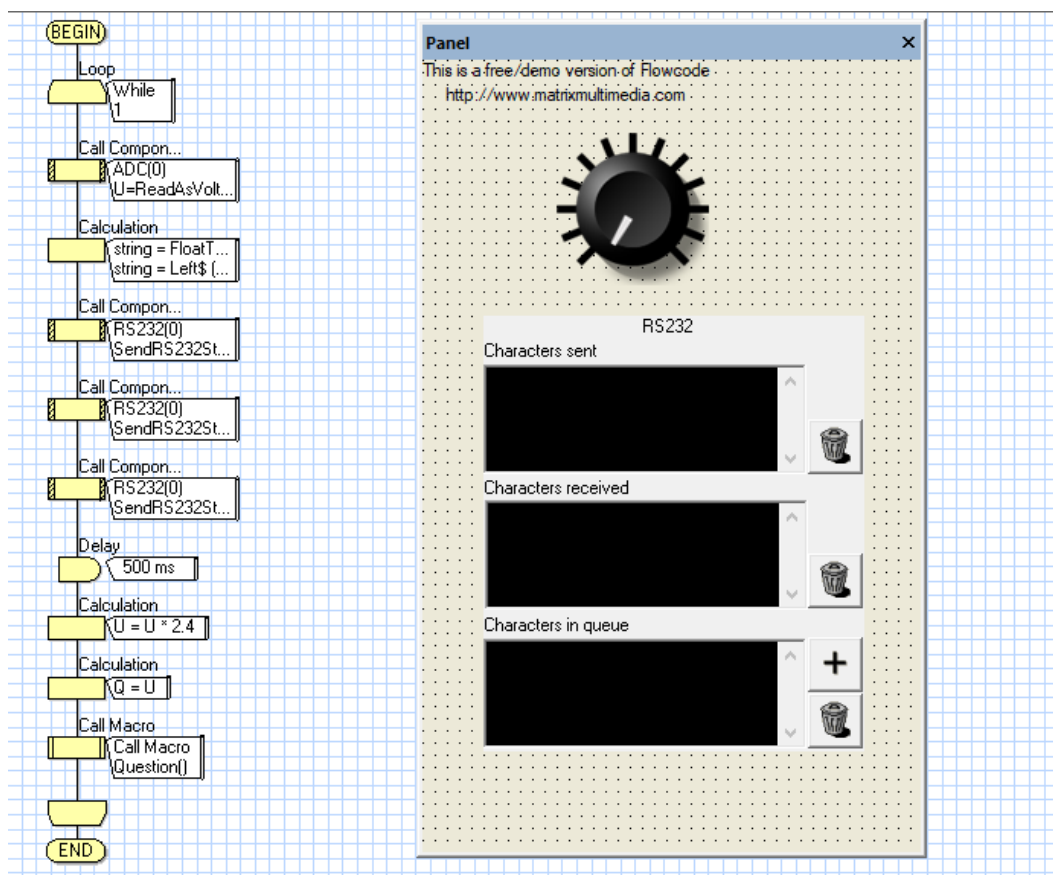


Рис. 2.59 – блок-схема

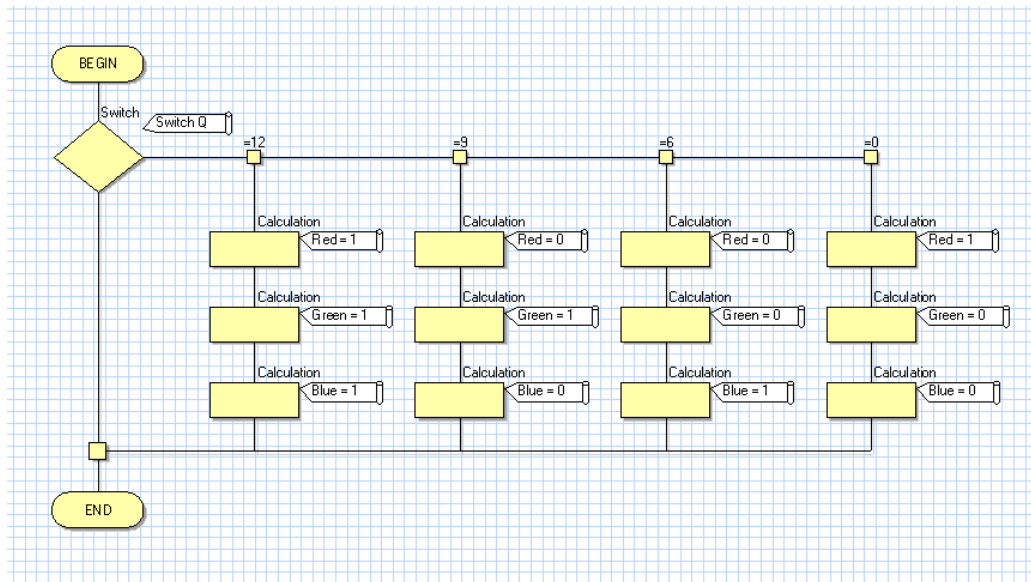


Рис. 2.60 – блок-схема для визначення кольору RGB

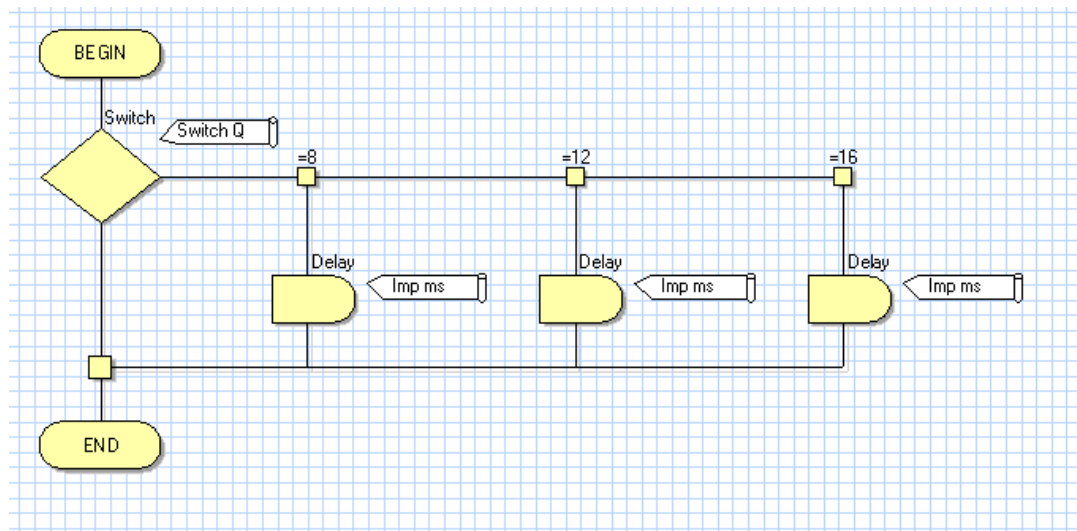


Рис. 2.61 – блок-схема для визначення амперів

2.2 Використані компоненти

Всі компоненти які були використані у побудуванні блок-схеми, докладно описані та зображені в наступних розділах.

2.2.2 Loop

У Flowcode, "loop" (або українською "цикл") є основною частиною програми, яка повторюється безкінечно. Це місце, де можна розмістити блоки коду, які будуть виконуватись знову і знову, доки програма працює.

Коли програма запускається, вона спочатку виконується один раз в рамках "циклу", а потім повторюється знову та знову, дозволяючи пристрою

постійно взаємодіяти зі світом навколо нього на основі налаштувань, умов та даних, вказаних у програмі.

Цикл "loop" у Flowcode дозволяє створювати програми, які мають постійний та безперервний режим роботи, виконуючи необхідні дії на основі програмних вказівок, доки програма не буде зупинена або перезавантажена.

На рис. 2.62 показаний цикл, надалі компоненти які нам потрібно розмістити поміщаємо в середину циклу [4].

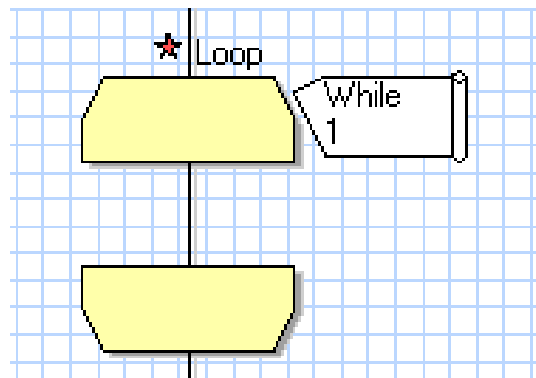


Рис. 2.62

2.2.3 Call Component Macro

"Call Component Macro" (виклик макроса компонента) є способом викликати певний фрагмент коду, який був написаний або визначений для певного компонента (або блоку) в програмі.

Коли створюєш складну програму у Flowcode, можуть виникнути ситуації, коли хочеш повторно використовувати певний блок коду або виконувати певні дії для кількох компонентів у програмі. Використання макросів дозволяє уникнути повторення коду та забезпечити більш організовану та модульну структуру програми.

За допомогою "Call Component Macro", можна створити макрос (або функцію) для певного компонента, який виконує певні дії або має певну логіку. Потім можна викликати цей макрос з інших частин програми, щоб виконати дії, пов'язані з цим макросом.

Виклик макроса може бути виконаний з будь-якого місця програми, де необхідно виконати певні дії або отримати результати, які забезпечуються

макросом компонента. Це дозволяє зробити код більш модульним, зрозумілим та легко змінюваним.

В цілому, "Call Component Macro" у Flowcode дозволяє використовувати макроси для організації та повторного використання коду, забезпечуючи більшу ефективність та гнучкість у розробці програмного забезпечення[4].

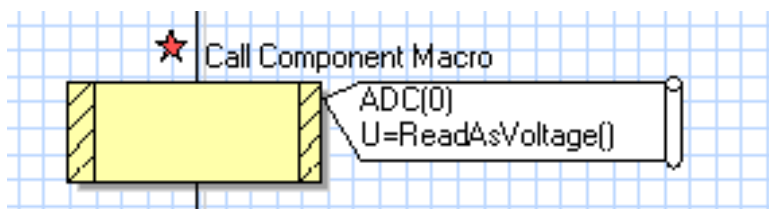


Рис. 2.63

В даному випадку на макрос компоненті який показаний на рис. 2.63 вибраний АЦП який показаний на рис. 2.64, макрос зчитує дані з АЦП та перетворює ці данні на змінну U з плаваючою точкою.



Рис. 2.64

Цей макрос компонент показаний на рис. 2.65 відображає змінну U на цифровій панелі і корегується згідно з АЦП, тобто при прокрутці АЦП дані на цифровій панелі постійно змінюються.

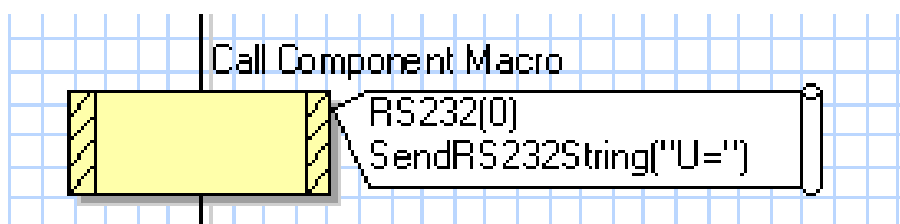


Рис. 2.65

Макрос компонент вказаний на рис. 2.66 показує строку на цифровій панелі де відображається змінна U.

У блоку Delay можна встановити час затримки у мілісекундах або використовувати величини часу, такі як секунди, хвилини тощо. Наприклад, якщо встановити блок Delay на 1000 мілісекунд, програма призупиниться на 1 секунду перед продовженням виконання наступних дій.

Затримки часто використовуються для керування таймерами, регулювання частоти опитування сенсорів або створення затримки між послідовними діями. Flowcode надає зручні блоки Delay, які дозволяють контролювати часові інтервали в твоїй програмі без необхідності вручну розраховувати затримки [4].

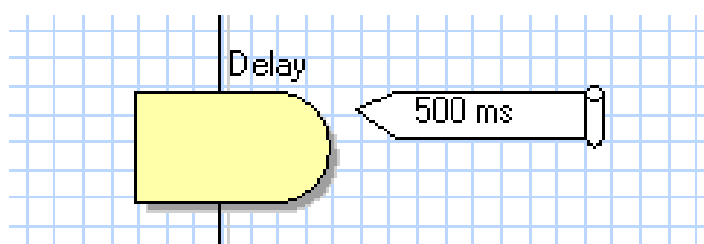


Рис. 2.71

2.2.6 Call Macro

"Call Macro" (виклик макроса) є способом викликати певний фрагмент коду, який був написаний або визначений як макрос у програмі.

Макрос - це набір команд або блоків коду, які можна використовувати багаторазово для виконання певних дій у програмі. Використання макросів дозволяє уникнути повторення коду та забезпечити більш організовану та модульну структуру програми.

Коли створюєш програму, можливо, знадобиться викликати певний фрагмент коду з різних частин програми. Використання блоку "Call Macro" дозволяє викликати макрос з будь-якого місця програми, де ти потребуєш виконати певні дії, що описані у макросі.

При виклику макросу, виконання програми переходить до визначеного макроса, виконує всі команди або блоки коду, які його складають, а потім повертається до місця виклику, продовжуючи виконання програми з того місця, де був зроблений виклик макросу.

Використання "Call Macro" дозволяє створювати більш модульні програми, де окремі фрагменти коду можуть бути використані багаторазово та легко змінюватись. Це сприяє ефективнішій розробці програмного забезпечення в Flowcode[4].

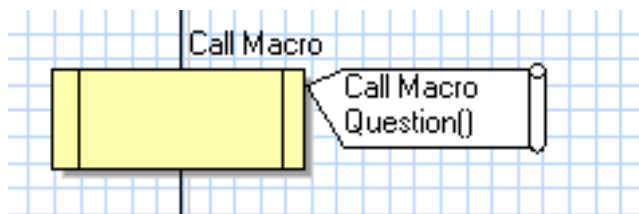


Рис. 2.72

В Call Macro робимо перехід в підпрограму де розташований Switch.

2.2.7 Switch

"Switch" (перемикач) - це блок або компонент, який дозволяє здійснювати рішення на основі різних умов або значень. Він використовується для керування виконанням різних гілок коду залежно від вибраного умови або значення.

Switch може бути використаний для реалізації конструкцій, таких як "case" або "if-else", де відбувається перевірка декількох умов і виконання відповідних дій. У Switch можна визначити набір умов або значень, і для кожного з них можна вказати, які дії повинні бути виконані.

Наприклад, якщо використовується числовий Switch, можна визначити кілька різних значень, і для кожного з них можна вказати певні дії, які мають бути виконані, якщо значення відповідає цій умові. При виконанні програми, Switch перевіряє значення або умови і переходить до відповідної гілки коду, де знаходиться вказана дія.

Switch може бути корисним для керування поведінкою програми на основі різних умов або значень, таких як вхідні дані, стан системи або рішення користувача. Він дозволяє створювати більш гнучкі та складні логічні структури у програмі Flowcode.

В Switch ставимо 4 вимоги, це 12/9/6/0 вольт, якщо покрутити АЦП на 5 вольт у нас Switch буде 12 вольт і виконається вимога та засвітиться біла

лампочка. Якщо 3,75 вольт, вимога буде 9 вольт та загориться зелена лампочка. На 2,5 вольт виконається вимога 6 вольт та загориться синя лампочка. Якщо на АЦП буде менше 2,5 вольт виконається вимога 0 вольт та загориться червона лампочка [4].

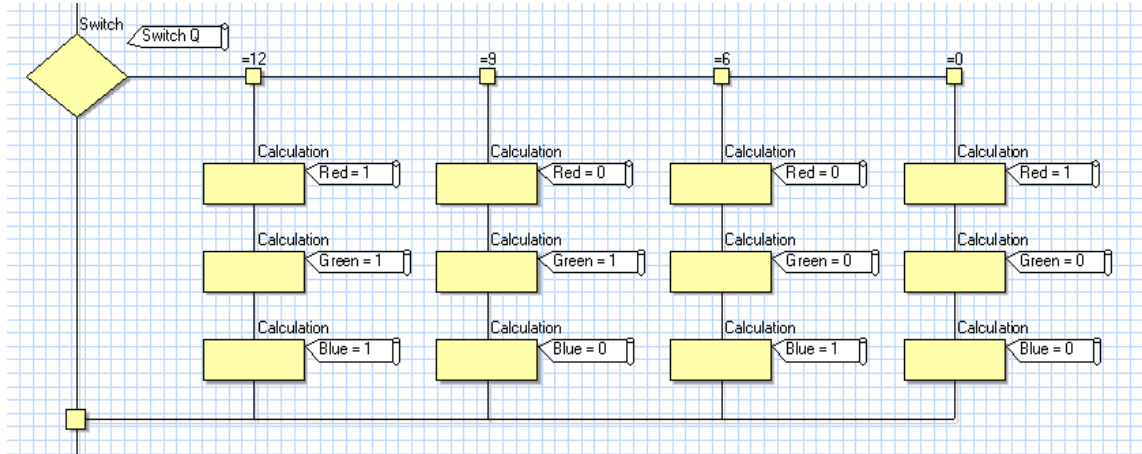


Рис. 2.73 – блок-схема для визначення кольору RGB

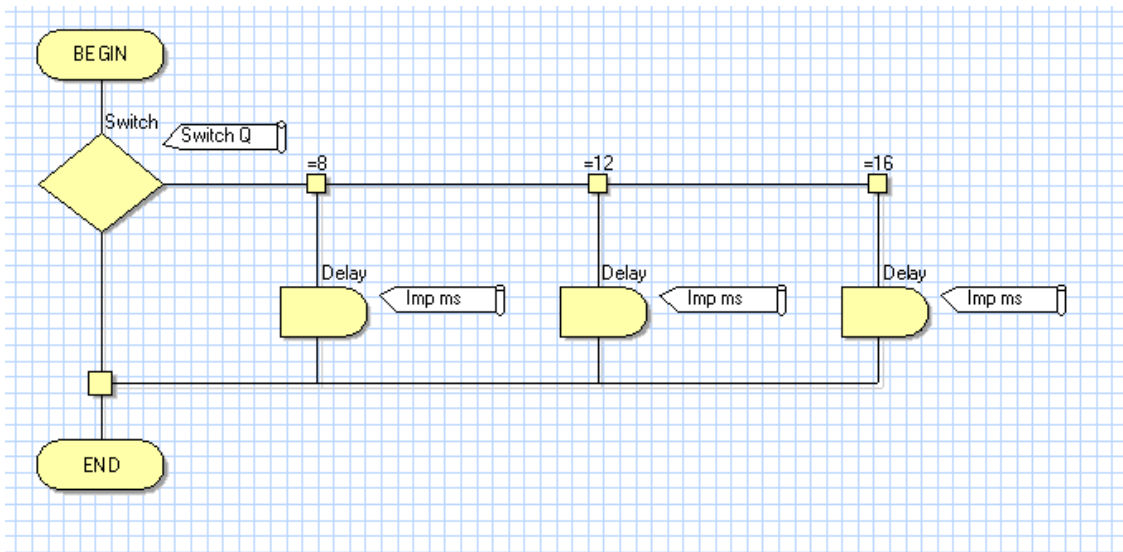


Рис. 2.74 – блок-схема для визначення амперів

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ПРОГРАМНО КЕРОВАНОГО КОМПЛЕКСУ СИСТЕМИ НА БАЗІ КОМП'ЮТЕРА ARDUINO

3.1 Розробка базової схеми

Використовуючи програмне забезпечення Fritzing, була розроблена базова схема підключення Arduino Uno до RGB світло діоду та потенціометра, схема підключення зображена на рис. 3.1 [7]

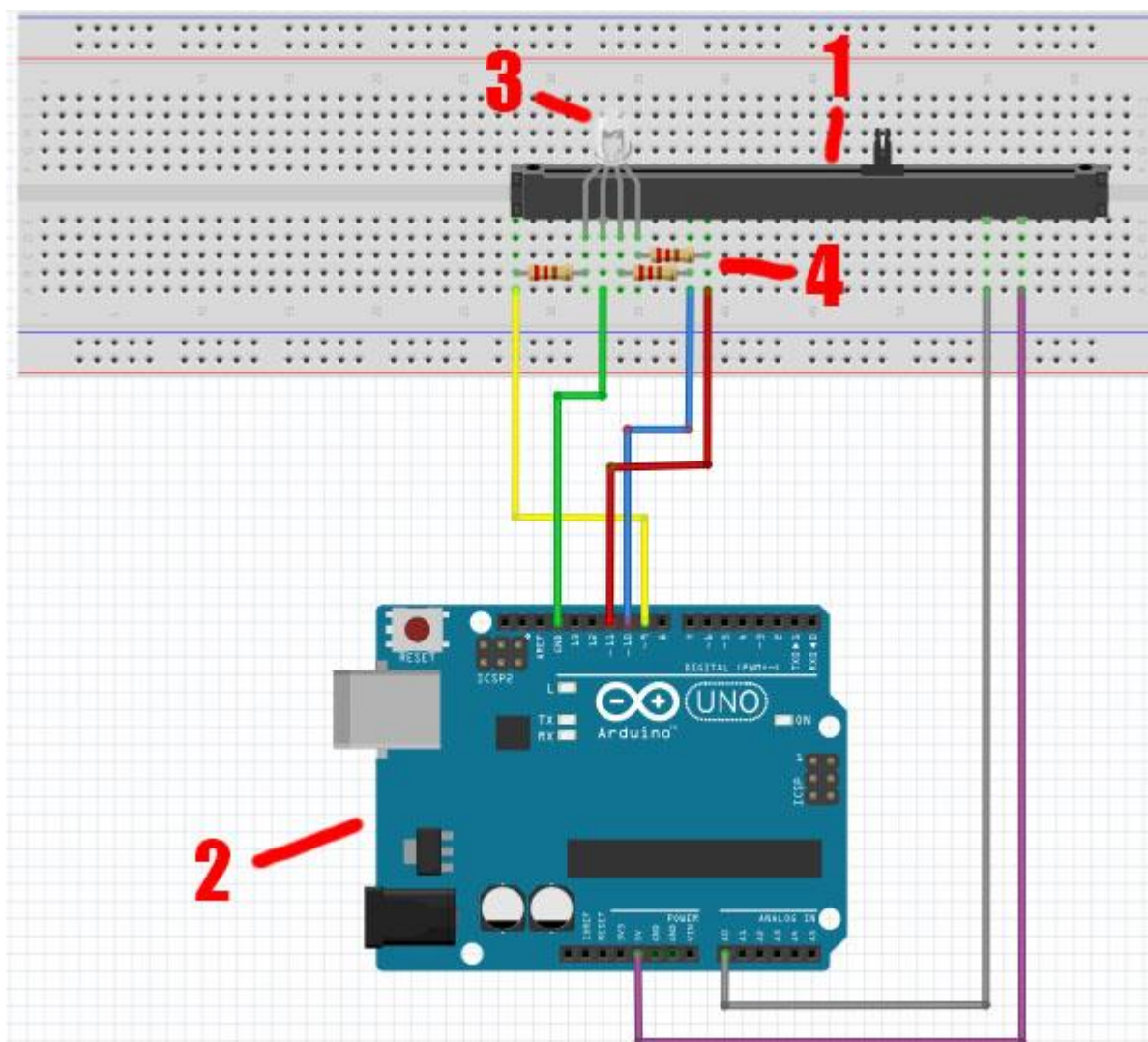


Рис. 3.1 – схема підключення Arduino Uno

На рис. 3.1 ми бачимо наступні елементи:

- 1) Потенціометр - це електричний компонент, який використовується для регулювання опору в електричному колі. В основі його роботи лежить змінювання опору шляхом руху спеціального повзуна (що може бути ручкою або іншим механізмом) по поверхні резистивного матеріалу.

Змн.	Арк.	№ докум.	Підпис	Дата

Потенціометри використовуються для регулювання напруги, струму або іншого параметру в електричних схемах. Один із загальних варіантів застосування - управління яскравістю світлодіодів, гучномовцями або іншими пристроями, де потрібно змінювати сигнал чи параметр. В даному випадку він змінює потенціал від 0-5 вольт.

2) Arduino Uno - це мікроконтролерна платформа, яка використовується для розробки прототипів електронних пристроїв. Вона має малий розмір, вбудований мікроконтролер ATmega328P, велику кількість цифрових та аналогових входів/виходів, інтерфейс USB для програмування та зчитування даних, а також можливість легко підключати різноманітні датчики та пристрої. Arduino Uno є популярним інструментом для електронних хобістів, студентів та професіоналів для створення прототипів і вивчення програмування мікроконтролерів [8].

3) RGB світлодіод (світлодіод із змінюваним кольором) - це тип світлодіода, який може випромінювати світло трьох основних кольорів: червоного (R), зеленого (G) і синього (B). Ці три кольори можна комбінувати в різноманітні співвідношення для отримання різних кольорів спектра.

RGB світлодіоди дозволяють створювати широкий спектр кольорів, включаючи білий, шляхом зміни інтенсивності і комбінацій основних кольорів. Це робить їх популярними для застосувань, де необхідне точне керування кольором, такі як освітлення в приміщеннях, світлові ефекти в розважальних заходах, відеоігри, та інші області.

Контроль над RGB світлодіодами може здійснюватися за допомогою мікроконтролерів або спеціальних контролерів, які дозволяють програмне встановлення кольорів та ефектів.

4) Резистор(FET) на 220 Ом - це резистор з опором 220 Ом на схемі їх зображено в кількості 3 одиниць. Опір вимірюється в омах (Ом) і вказує, як легко або важко електричний струм проходить через резистор. У випадку резистора на 220 Ом опір становить 220 Ом.

Резистори використовуються для обмеження струму в електричних ланцюгах, поділу напруги, а також для інших цілей в електронних схемах. Значення опору резистора визначається його конструкцією та матеріалом, і в даному випадку це 220 Ом, що є стандартним значенням для багатьох застосувань.

3.2 Розробка блок-схеми напруги пілот сигналу

Після цього, була розроблена блок-схема напруги пілот сигналу яка вказана на рис. 3.2

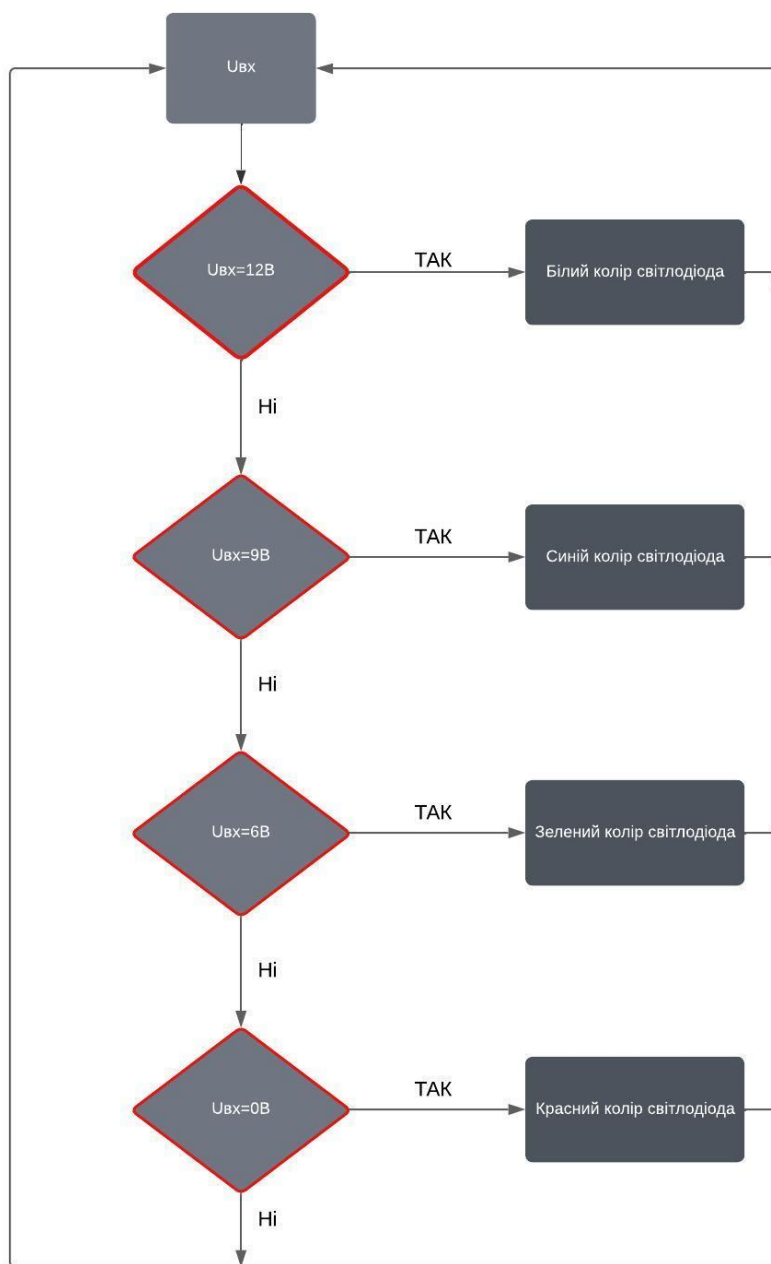


Рис. 3.2 – блок-схема напруги пілот сигналу

В ній зображені наступні дії, якщо сигнал відповідає 12В, то світло діод буде світитись білим кольором, якщо сигнал не відповідає 12В, то дії переходять далі, зображено на рис. 3.3

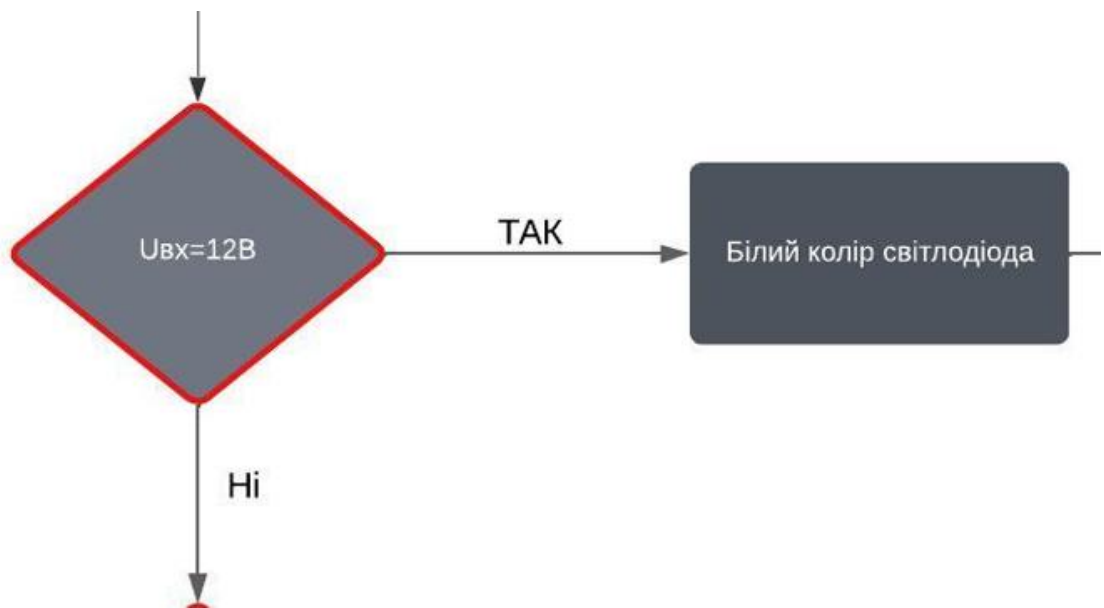


Рис. 3.3 – частина блок-схеми на 12В

Потім в блок схемі виконуються схожі дії, тільки вже для 9В, якщо сигнал відповідає 9В, то світло діод буде світитись синім кольором, якщо сигнал не відповідає 9В, дії переходять далі, зображено на рис. 3.4

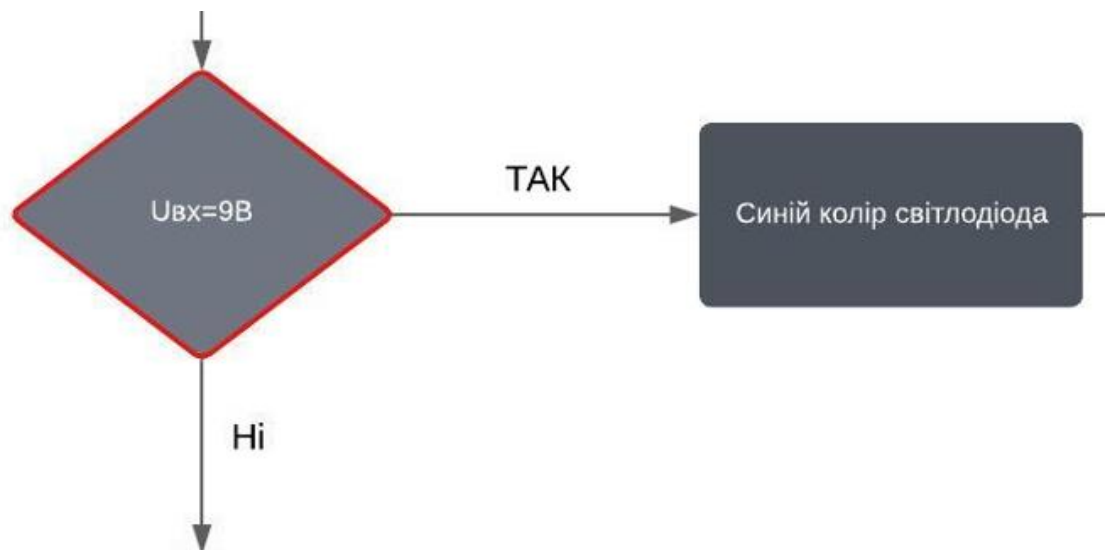


Рис. 3.4 – частина блок-схеми на 9В

Далі блок схема переходить до 6В, якщо сигнал відповідає 6В, то світло діод світиться зеленим кольором, якщо сигнал не відповідає 6В, дії переходять далі, зображено на рис. 3.5

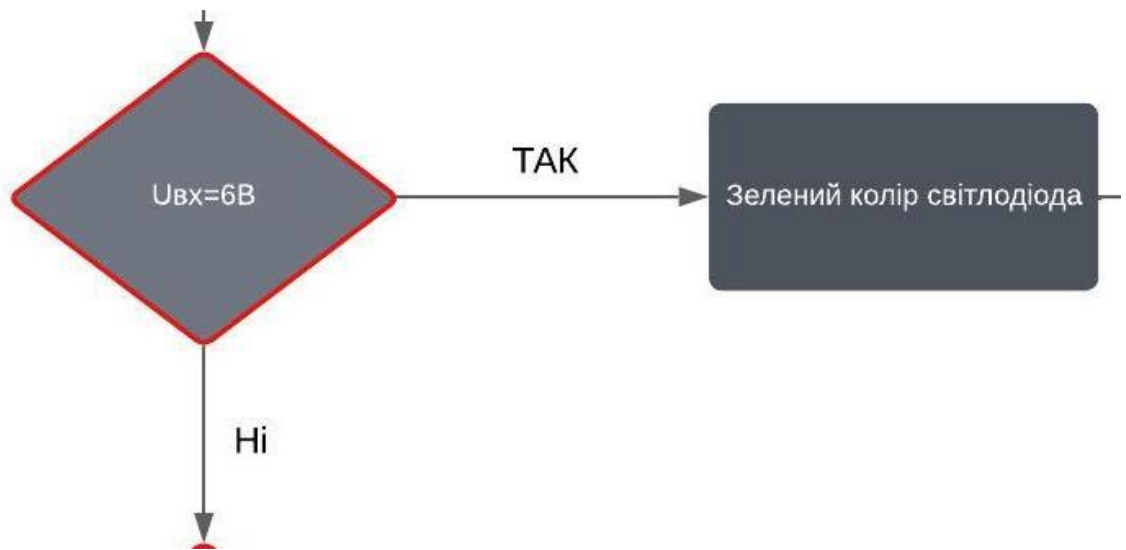


Рис. 3.5 – частина блок-схеми на 6В

Після цього блок-схема переходить до 0В, якщо сигнал відповідає 0В, то світло діод світиться червоним кольором, якщо ж сигнал не відповідає 0В, в такому випадку дії переходять на початок блок-схеми, вказано на рис. 3.6

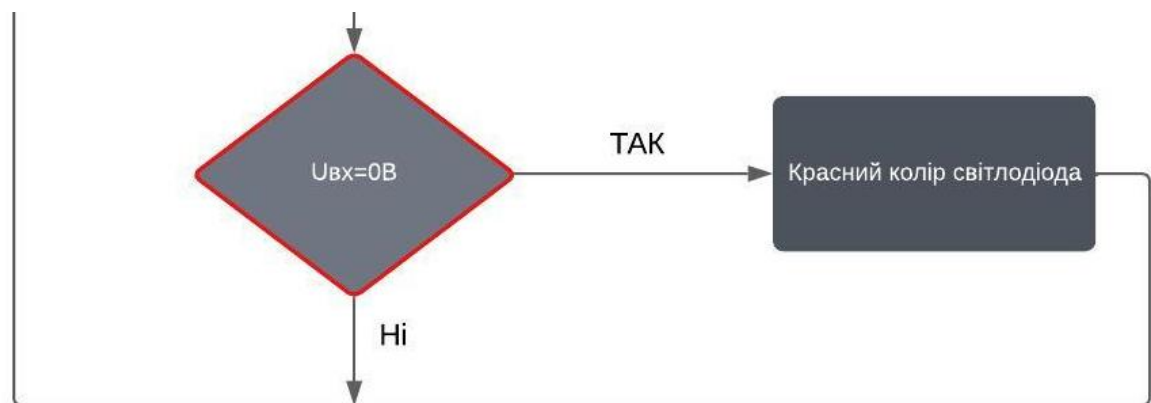


Рис. 3.6 – частина блок-схеми на 0В

3.3 Розробка блок-схеми значення струмової установки

Також була розроблена блок-схема значення струмової установки, вказана на рис. 3.7

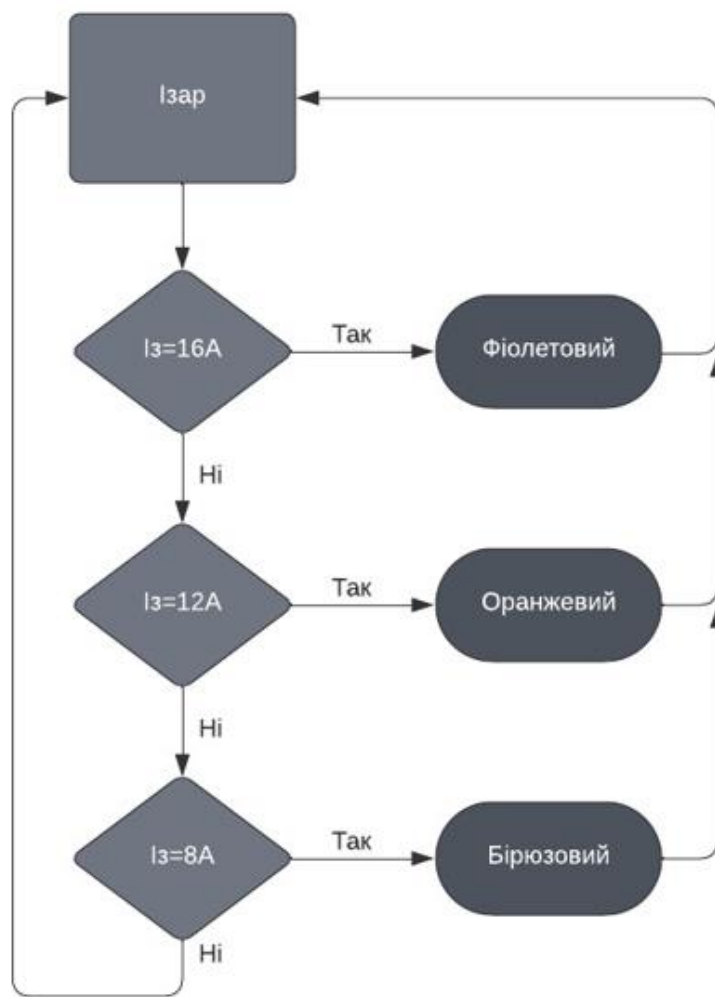


Рис. 3.7 – блок-схема значень струмової установки

В цій блок-схемі зображені наступні дії, якщо сигнал відповідає 16А, то колір світло діода буде фіолетовим, якщо сигнал не відповідає 16А, то дії переходять до наступного блоку, зображено на рис. 3.8

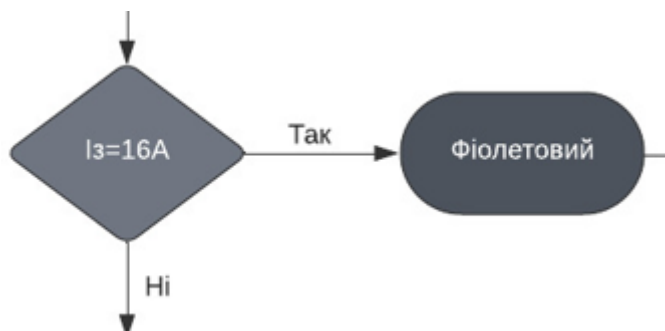


Рис. 3.8 – частина блок-схеми на 16А

Далі блок-схема переходить до 12А, якщо сигнал відповідає 12А, то світло діод буде світитись оранжевим кольором, якщо не відповідає, то дії переходять до наступного блоку, показано на рис. 3.9

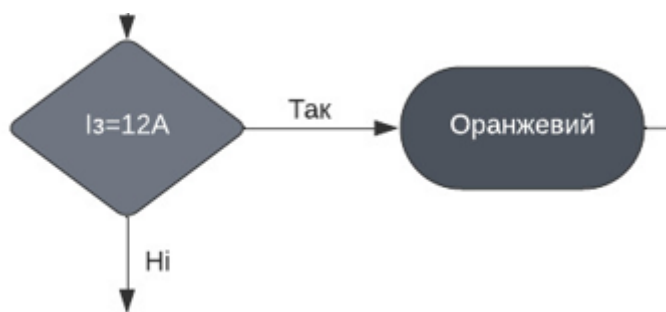


Рис. 3.9 – частина блок-схеми на 12А

Після цього блок-схема переходить до 8А, якщо сигнал відповідає 8А, то світло діод буде бірюзовим кольором, якщо не відповідає 8А, то блок-схема переходить на початок і повторює ці процеси, вказано на рис. 3.10

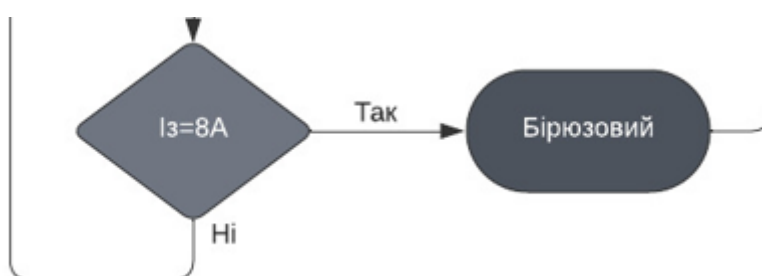


Рис. 3.10 – частина блок-схеми на 8А

3.4 Написання коду в Arduino IDE

Після побудування блок-схеми, був написаний код, за допомогою програмного забезпечення Arduino IDE.

Arduino IDE (Integrated Development Environment) - це середовище інтегрованої розробки для програмування мікроконтролерів на платформі Arduino. Це програмне забезпечення, яке надає зручний інтерфейс для написання, завантаження та відлагодження програм для Arduino.

Основні функції Arduino IDE включають текстовий редактор для написання програмних кодів мовою Arduino, можливість компіляції програми у виконуваний код, завантаження програми на плату Arduino через USB-порт, а також інструменти для відлагодження та моніторингу виведення пристрою.

Arduino IDE спрощує процес розробки для початківців, але також надає повний доступ до коду для більш досвідчених користувачів, щоб вони могли


```

22
23 // Зміна кольору світлодіода в залежності від напруги
24 if (potValue >= 682) { // 5 вольт або більше
25     analogWrite(redPin, 255);
26     analogWrite(greenPin, 255);
27     analogWrite(bluePin, 255);
28 } else if (potValue >= 512) { // від 3.75 вольт до 5 вольт
29     analogWrite(redPin, 0);
30     analogWrite(greenPin, 0);
31     analogWrite(bluePin, 255);
32 } else if (potValue >= 341) { // від 2.5 вольт до 3.75 вольт
33     analogWrite(redPin, 0);
34     analogWrite(greenPin, 255);
35     analogWrite(bluePin, 0);
36 } else { // менше 2.5 вольт
37     analogWrite(redPin, 255);
38     analogWrite(greenPin, 0);
39     analogWrite(bluePin, 0);
40 }
41
42 // Затримка для плавних переходів
43 delay(10);
44 }
45

```

Рис. 3.13 – друга частина коду

Також був написаний код для блок-схеми значення струмової установки вказаний на рис. 3.14, та продовження коду на рис. 3.15 і рис. 3.16

```

1 // Підключення виводів світлодіода і потенціометра
2 const int redPin = 9;
3 const int greenPin = 10;
4 const int bluePin = 11;
5 const int potPin = A0;
6
7 void setup() {
8     // Встановлення виводів світлодіода як виводів OUTPUT
9     pinMode(redPin, OUTPUT);
10    pinMode(greenPin, OUTPUT);
11    pinMode(bluePin, OUTPUT);
12 }
13
14 void loop() {
15     // Зчитування значення потенціометра
16     int potValue = analogRead(potPin);
17
18     // Розбиття значення потенціометра на три частини
19     int redValue, greenValue, blueValue;
20

```

Рис. 3.14 – перша частина коду

```

20
21 // Зміна кольору світлодіода в залежності від напруги
22 if (potValue >= 682) { // 5 вольт або більше
23     redValue = 255;
24     greenValue = 255;
25     blueValue = 255; // білий
26 } else if (potValue >= 614) { // від 4.5 вольт до 5 вольт
27     redValue = 128;
28     greenValue = 0;
29     blueValue = 128; // фіолетовий
30 } else if (potValue >= 568) { // від 4 вольт до 4.5 вольт
31     redValue = 255;
32     greenValue = 165;
33     blueValue = 0; // оранжевий
34 } else if (potValue >= 511) { // від 3.5 вольт до 4 вольт
35     redValue = 0;
36     greenValue = 255;
37     blueValue = 255; // бірюзовий
38 } else if (potValue >= 468) { // від 3 вольт до 3.5 вольт
39     redValue = 75;
40     greenValue = 0;
41     blueValue = 130; // фіолетовий
42 } else if (potValue >= 409) { // від 2.7 вольт до 3 вольт
43     redValue = 255;
44     greenValue = 69;
45     blueValue = 0; // оранжевий

```

Рис. 3.15 – друга частина коду

```

45     blueValue = 0; // оранжевий
46 } else if (potValue >= 341) { // від 2.6 вольт до 2.7 вольт
47     redValue = 0;
48     greenValue = 238;
49     blueValue = 238; // бірюзовий
50 } else { // менше 2.6 вольт
51     redValue = 0;
52     greenValue = 0;
53     blueValue = 0; // вимкнений
54 }
55
56 // Зміна кольору світлодіода
57 analogWrite(redPin, redValue);
58 analogWrite(greenPin, greenValue);
59 analogWrite(bluePin, blueValue);
60
61 // Затримка для плавних переходів
62 delay(10);
63 }
64

```

Рис. 3.16 – третя частина коду

3.5 Завантаження коду на плату Arduino

Після написання коду, перевіривши його в самій програмі, його було завантажено на плату Arduino Uno, на рисунках нижче буде видно роботу RGB світло діода відповідно до написаного коду, а саме напруги пілот сигналу.

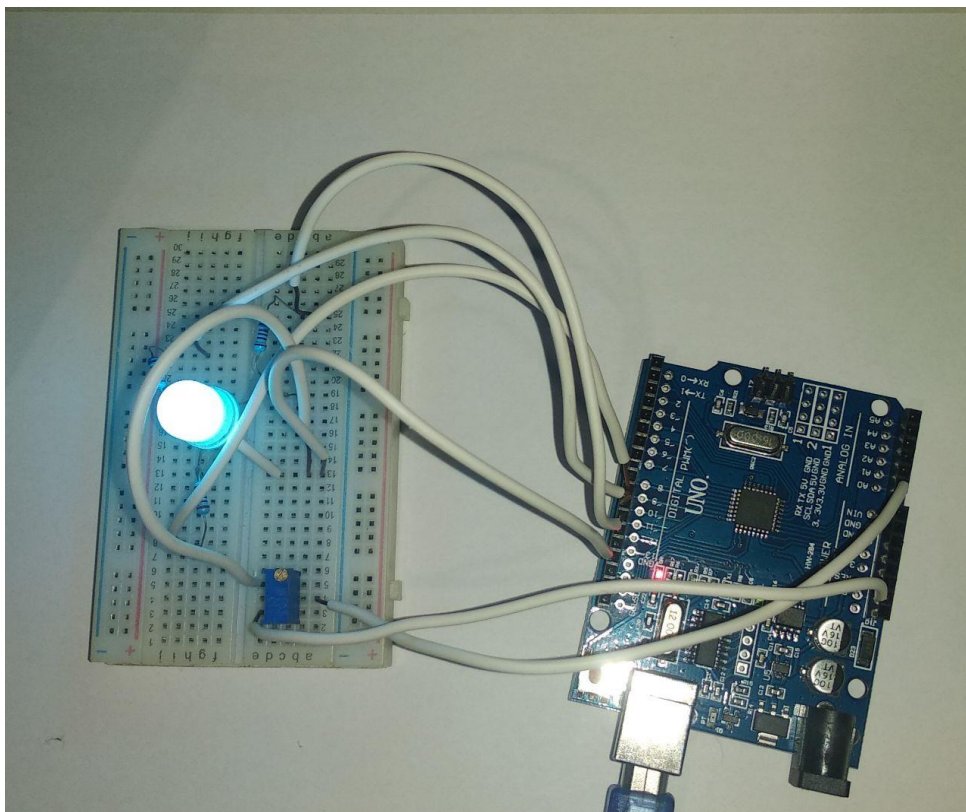


Рис. 3.17 – подача 5В, RGB світло діод білого кольору

Змн.	Арк.	№ докум.	Підпис	Дата

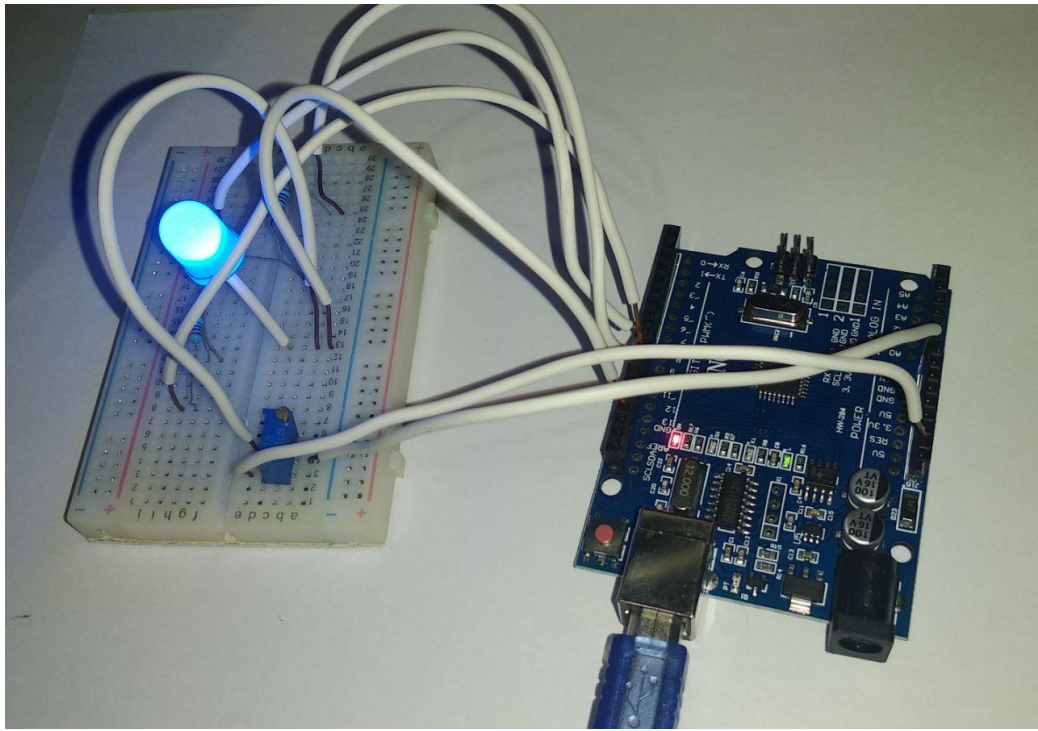


Рис. 3.18 – подача 3,75В, RGB світло діод синього кольору

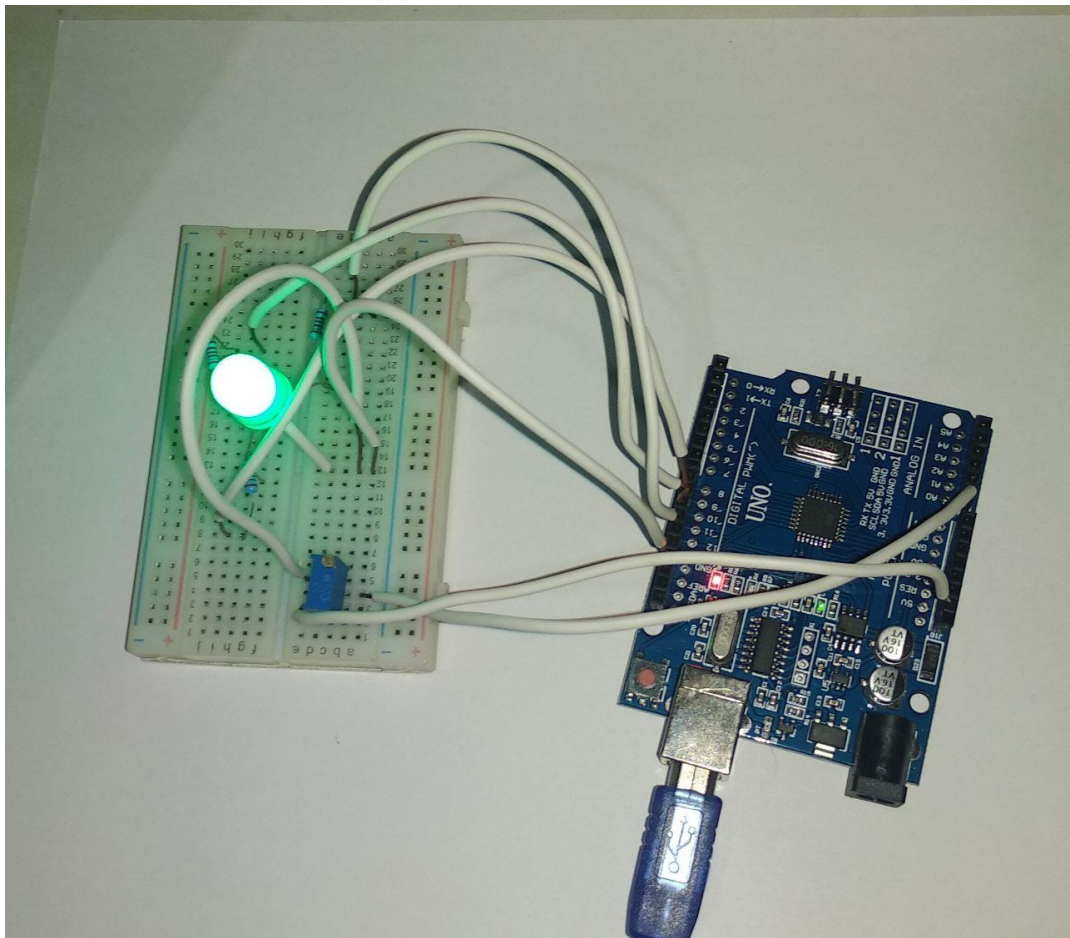


Рис. 3.19 – подача 2,5В, RGB світло діод зеленого кольору

Змн.	Арк.	№ докум.	Підпис	Дата

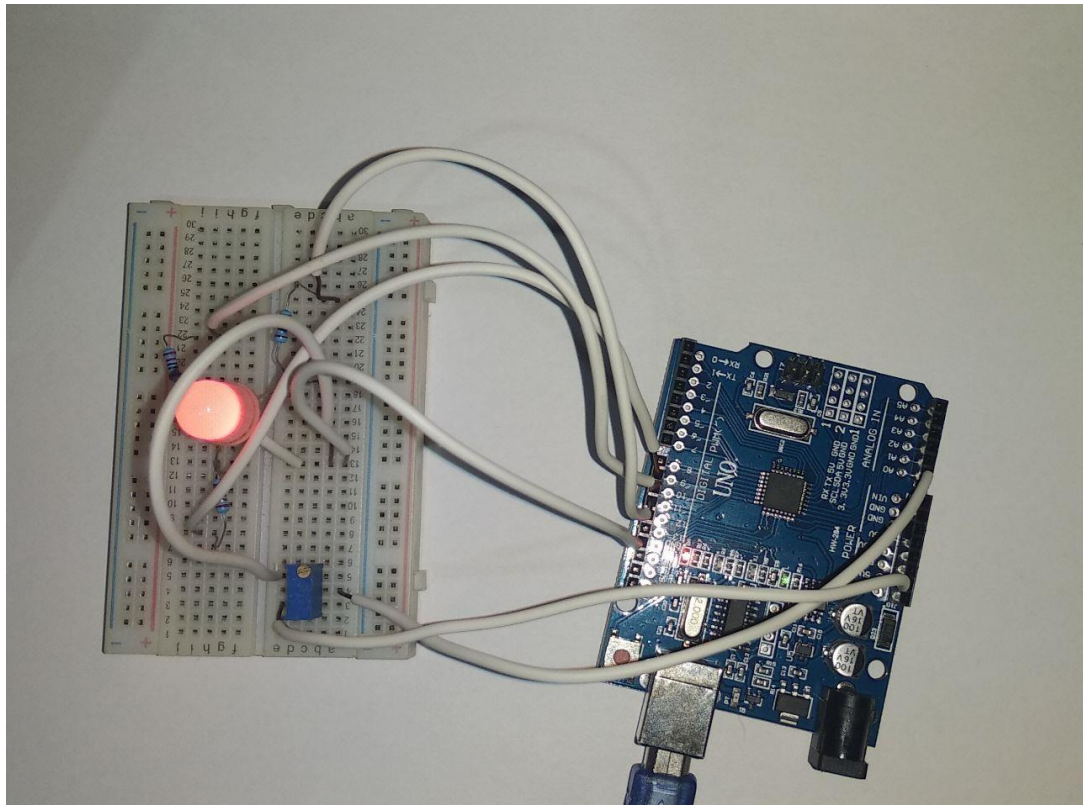


Рис. 3.20 – подача нижче 2,5В, RGB світло діод червоного кольору
 Робота RGB світло діоду відповідно до другого написання коду, а саме значення струмової установки.

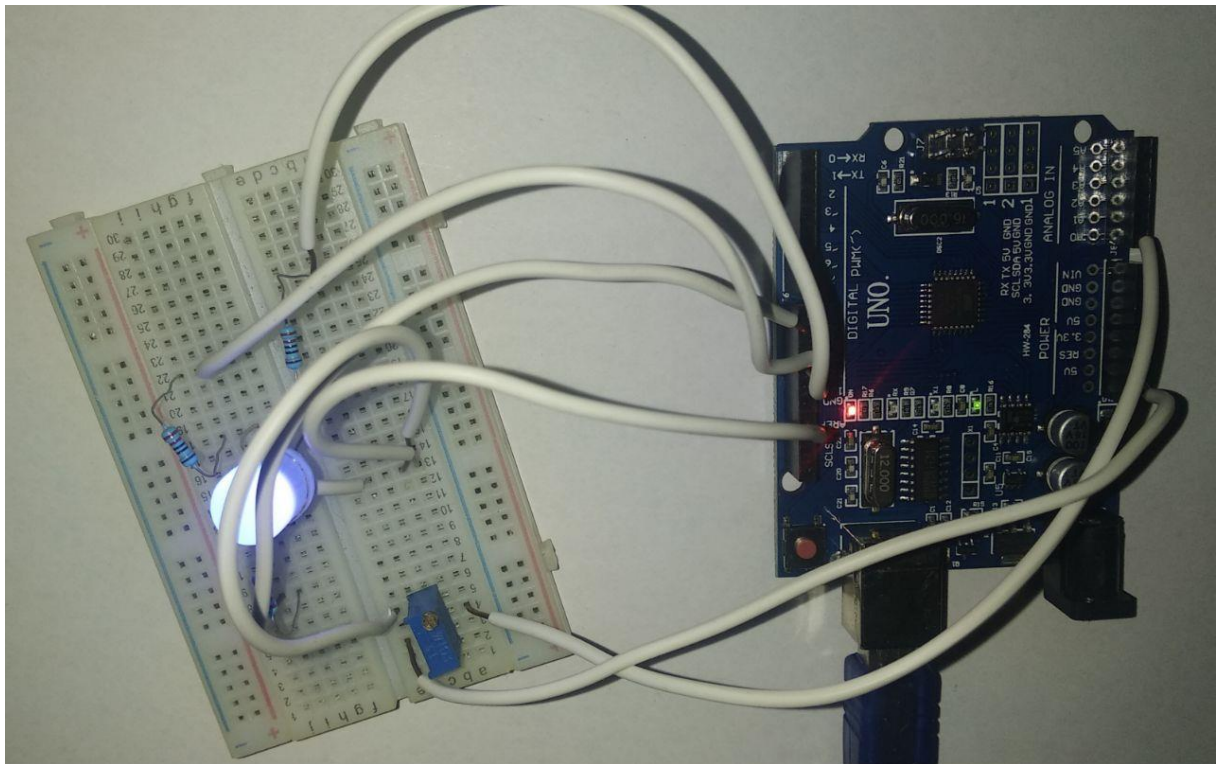


Рис. 3.21 – подача 4,5В, RGB світло діод фіолетового кольору

Змн.	Арк.	№ докум.	Підпис	Дата

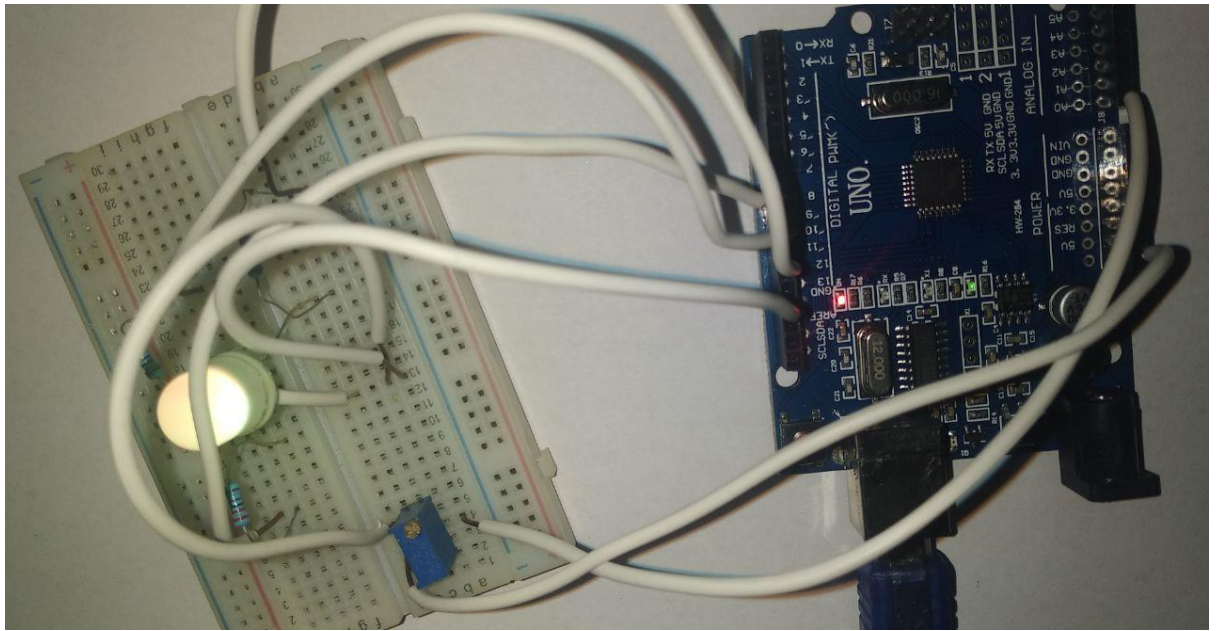


Рис. 3.22 – подача 4В, RGB світло діод помаранчевого кольору

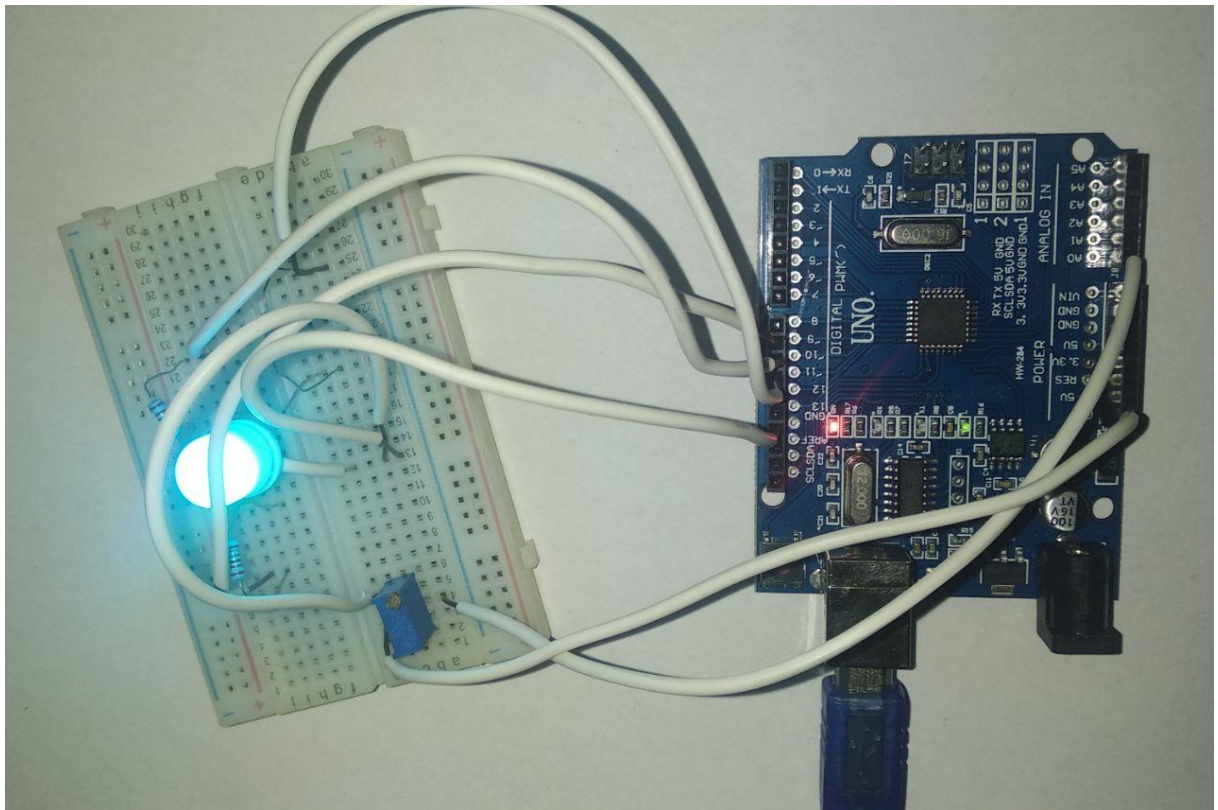


Рис. 3.23 – подача 3,5В, RGB світло діод бірюзового кольору

3.6 Заміри напруги

Після цього перейшли до замірів напруги, щоб впевнитись в коректній роботі плати, та в правильному написанні коду, для напруги пілот сигналу.

Змн.	Арк.	№ докум.	Підпис	Дата

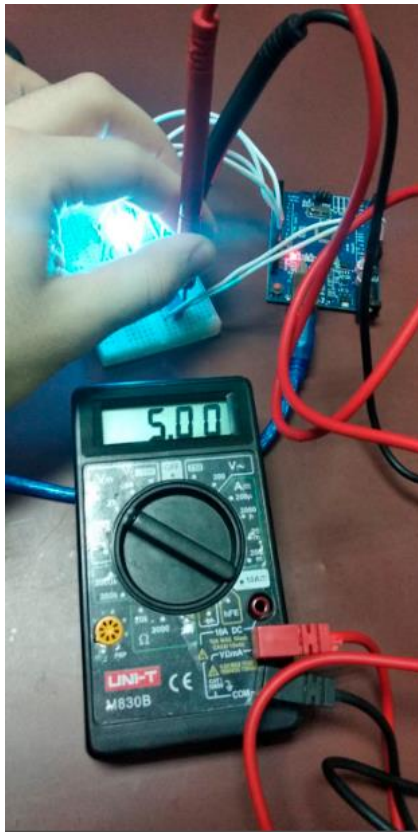


Рис. 3.24 – вимір напруги на світло діоді білого кольору



Рис. 3.25 – вимір напруги на світло діоді синього кольору

					7.141.190034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		61

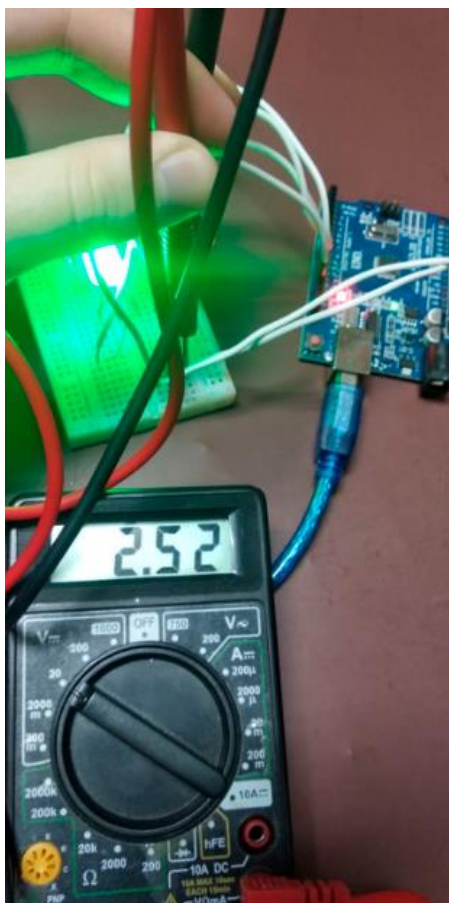


Рис. 3.26 – вимір напруги на світло діоді зеленого кольору

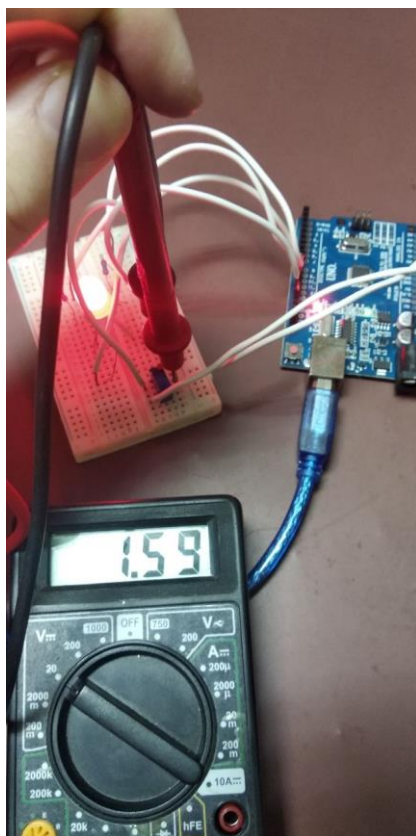


Рис. 3.27 – вимір напруги на світло діоді червоного кольору

					7.141.190034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		62

Потім повторили заміри напруги, щоб впевнитись в коректній роботі плати, та в правильному написанні коду для значення струмової установки.

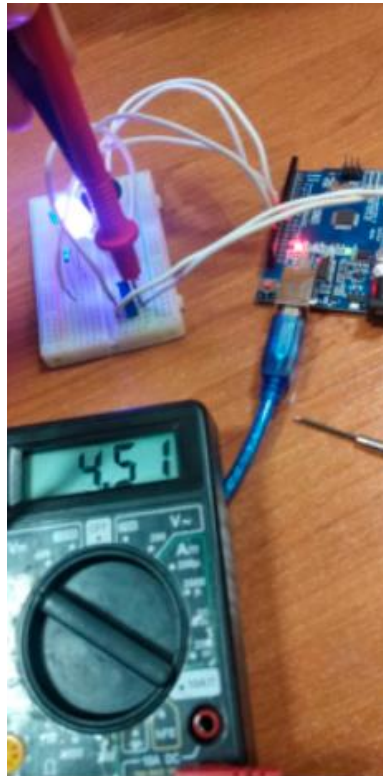


Рис. 3.28 – вимір напруги на світло діоді фіолетового кольору



Рис. 3.29 – вимір напруги на світло діоді помаранчевого кольору

					7.141.190034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		63

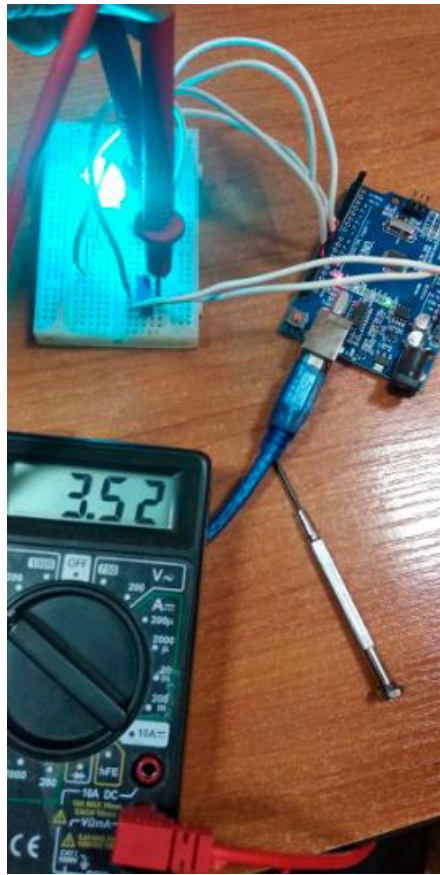


Рис. 3.30 - вимір напруги на світло діоді бірюзового кольору

					7.141.190034.ПЗ	Арк.
						64
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 4

ВИМОГИ ДО КВАЛІФІКАЦІЙНИХ ГРУП ПО ЕЛЕКТРОБЕЗПЕЦІ ТА ПОРЯДОК ОТРИМАННЯ КВАЛІФІКАЦІЙНИХ ГРУП

4.1 Вимоги до працівників та їх підготовки

Основні вимоги до працівників щодо проведення навчання і перевірки знань з питань охорони праці відповідно до вимог ГНД 34.12.102-2004, інструктаж із питань пожежної безпеки, спеціальне навчання та перевірка знань з питань пожежної безпеки, а також перелік посад, які зобов'язані проходити навчання та перевірку знань з питань пожежної безпеки та порядок їх організації визначаються НПАОП 0.00-4.12-05, НАПБ Б.02.005-2003, при виконанні робіт в електроустановках, розміщених у вибухонебезпечних або пожежонебезпечних зонах, згідно з вимогами НАПБ Б.06.001-2003 [9].

Обслуговування електроустановок споживачів, у тому числі виконання ремонтних, монтажних, налагоджувальних робіт і оперативних перемикачів в електроустановках, повинні здійснювати спеціально підготовлені електротехнічні працівники, а саме: адміністративно-технічні, оперативні, виробничі та оперативно-виробничі працівники споживача або працівники спеціалізованої організації (за договором) [9].

Обслуговування установок електротехнологічних процесів (електрозварювання, електролізу, електротермії тощо), вантажопідіймальних механізмів, ручних електричних машин, переносних та пересувних струмоприймачів, складного енергонасиченого виробничо-технологічного обладнання, під час роботи якого необхідно постійно проводити технічний нагляд, зміну, коригування ведення технологічних режимів за допомогою штатних засобів регулювання електроапаратури, електроприводів, повинні здійснювати спеціально підготовлені електротехнологічні працівники, які мають навички та знання для безпечного виконання робіт з технічного обслуговування закріпленої за ним установки [9].

					7.141.190034.ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		65

Електротехнологічні працівники виробничих цехів та дільниць, які здійснюють експлуатацію електротехнологічних установок, повинні мати групу з електробезпеки не нижче другої.

Керівники структурних підрозділів, яким безпосередньо підпорядковані електротехнологічні працівники, повинні мати групу з електробезпеки не нижчу, ніж у підлеглих працівників. Вони повинні здійснювати технічне керівництво цими працівниками та контроль за їхньою роботою [9].

Працівники, які обслуговують електроустановки споживачів або технологічні процеси, які базуються на використанні електричної енергії, повинні мати вік понад 18 років.

При прийнятті на роботу, а також періодично під час роботи стан здоров'я цих працівників повинен засвідчуватися медичним оглядом.

Керівник споживача відповідно до вимог ГНД 34.12.102-2004 з урахуванням енергоємності та складу енергетичного обладнання повинен затвердити відповідні положення про спеціальну підготовку і навчання електротехнічних та електротехнологічних працівників з питань технічної експлуатації електроустановок споживачів[9].

У процесі трудової діяльності електротехнічні та електротехнологічні працівники проходять відповідно до вимог ГНД 34.12.102-2004 такі форми навчання з питань технічної експлуатації електроустановок:

- інструктажі;
- періодичне навчання спеціалізованих навчальних закладах (далі – СНЗ);
- щорічне навчання на підприємстві;
- спеціальну підготовку;
- стажування;
- дублювання;
- протиаварійні тренування.

Посвідчення про перевірку знань працівника є документом, який засвідчує право на самостійну роботу в електроустановках на зазначеній посаді за фахом [10].

Посвідчення про перевірку знань видається працівникові комісією з перевірки знань підприємства, організації після перевірки знань і є дійсним тільки після внесення відповідних записів [10].

Під час виконання службових обов'язків працівник повинен мати з собою посвідчення про перевірку знань. За відсутності посвідчення або за наявності посвідчення з простроченими термінами перевірки знань працівник до роботи не допускається [10].

Посвідчення про перевірку знань підлягає заміні у випадку зміни посади або за відсутності місця для записів.

Посвідчення про перевірку знань вилучається у працівника комісією з перевірки знань в разі незадовільних знань, керівником структурного підрозділу – в разі вигасання терміну дії медичного огляду [10].

					7.141.190034.ПЗ	Арк.
						69
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Використовуючи середовище Flowcode були розроблені: принципова блок-схема для визначення кольору RGB та блок-схема для визначення амперів.

За допомогою середовища Arduino IDE був написаний код для блок-схеми напруги пілот сигналу, також був написаний код для блок-схеми значення струмової установки.

Зібравши схему та завантаживши на неї написаний код, ми перевірили її працездатність і коректність.

					7.141.190034.ПЗ	Арк.
						70
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ГРАФІЧНИХ РОБІТ

- 1) Розробка принципової блок-схеми.
- 2) Розробка базової схеми підключення Arduino Uno.
- 3) Розробка блок-схеми напруги пілот сигналу та значення струмової установки.
- 4) Написання коду в Arduino IDE.
- 5) Завантаження коду на плату Arduino UNO.

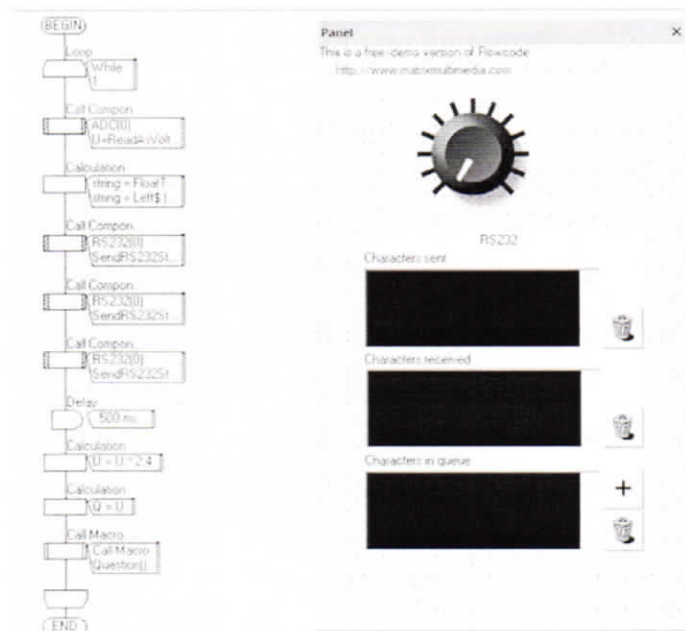
Заміри напруги на платі.

					7.141.190034.ПЗ	Арк.
						72
Змн.	Арк.	№ докум.	Підпис	Дата		

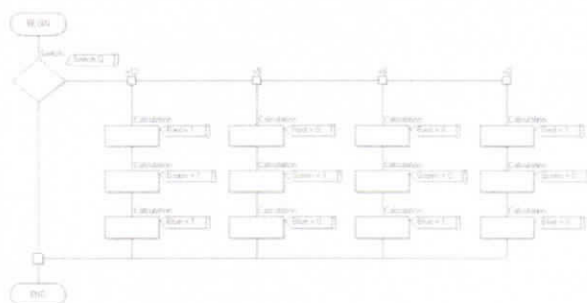
ДОДАТОК А

					7.141.190034.ПЗ	Арк.
						73
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

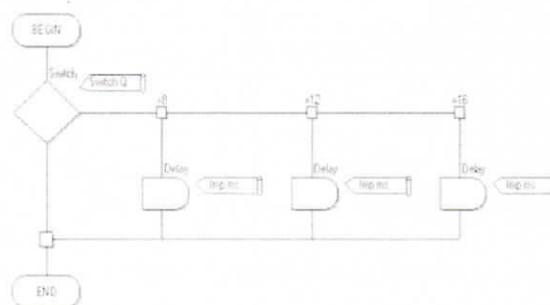
РОЗРОБКА ПРИНЦИПОВОЇ БЛОК-СХЕМИ



Принципова блок-схема



Блок-схема для визначення кольору RGB



Блок-схема для визначення амперів

					Розробка принципової блок-схеми		
Змн.	Арк.	№ докум.	Підпис	Дата	Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля		
					Літ.	Маса	Масштаб
Розроб.		Васильковський	<i>[Signature]</i>	20.12		1	1 : 1
Перевір.		Муха.А.М.	<i>[Signature]</i>	24.12	Арк.	74	Аркушів 79
Т. Контр.					МОН України. УДУНТ		
Реценз.					Кафедра «ЕТЕМ»		
Н. Контр.		Карзова О.О.	<i>[Signature]</i>	21.12	7.141.190034.01		
Затверд.		Муха А.М.	<i>[Signature]</i>	24.12			

РОЗРОБКА БАЗОВОЇ СХЕМИ ПІДКЛЮЧЕННЯ Arduino UNO

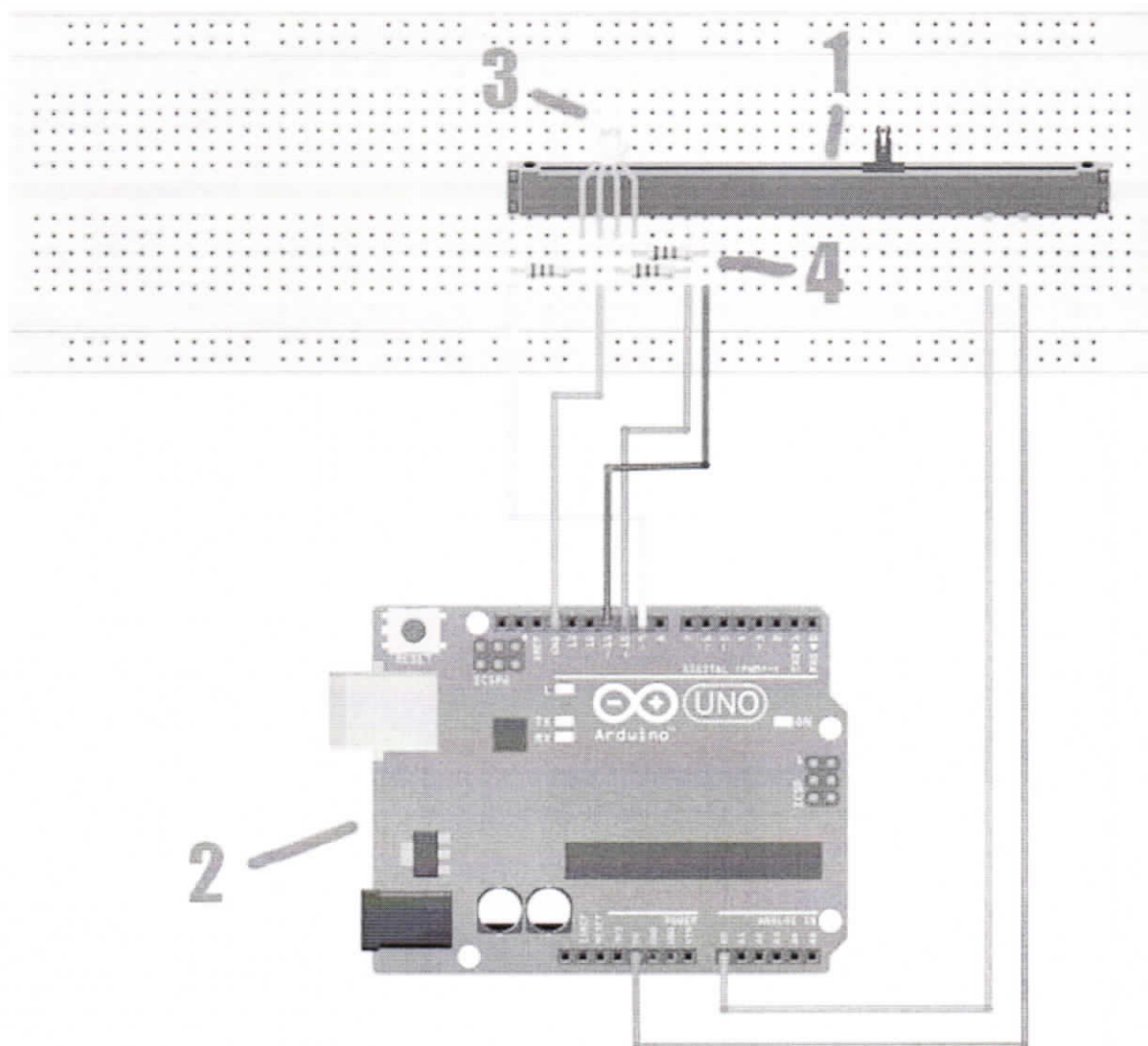
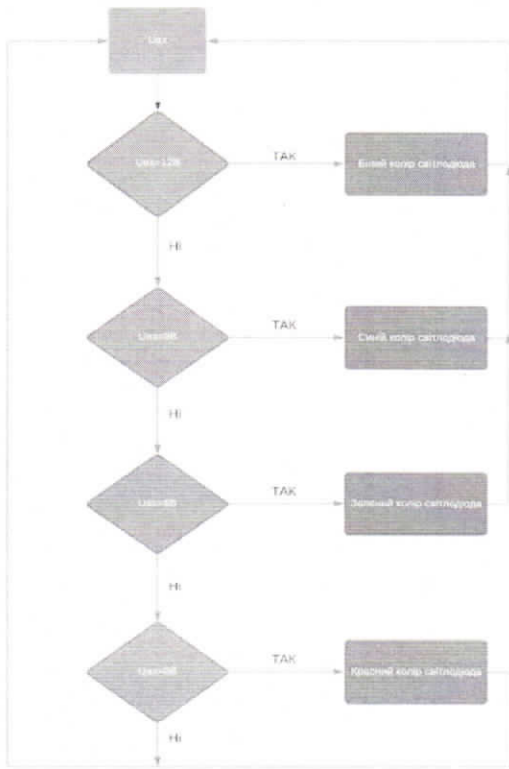


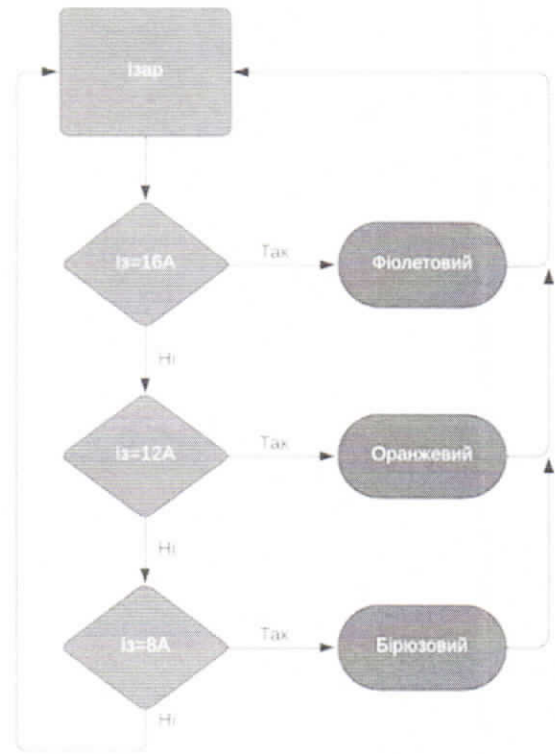
Схема підключення Arduino Uno

					Розробка базової схеми підключення Arduino UNO				
						Літ.	Маса	Масштаб	
Змін.	Арк.	№ докум.	Підпис	Дата	Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля			1	1 : 1
Розроб.		Васильковський	<i>[Signature]</i>	20.12	Арк.	75	Аркушів	79	
Перевір.		Муха.А.М.	<i>[Signature]</i>	2015	МОН України. УДУНТ				
Т. Контр.					Кафедра «ЕТЕМ»				
Реценз.					7.141.190034.02				
Н. Контр.		Карзова О.О.	<i>[Signature]</i>	2012					
Затверд.		Муха А.М.	<i>[Signature]</i>	2011					

РОЗРОБКА БЛОК-СХЕМИ НАПРУГИ ПІЛОТ СИГНАЛУ ТА ЗНАЧЕННЯ СТРУМОВОЇ УСТАНОВКИ



Напряга пілот сигналу



Значення струмової установки

					Розробка блок-схеми напруги пілот сигналу та значення струмової установки		
					Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля		
Змн.	Арк.	№ докум.	Підпис	Дата	Літ.	Маса	Масштаб
Розроб.		Васильковський	<i>[Signature]</i>	20.12		1	1 : 1
Перевір.		Муха.А.М.	<i>[Signature]</i>	20.12			
Т. Контр.					Арк.	76	Аркушів 79
Реценз.					МОН України. УДУНТ		
Н. Контр.		Карзова О.О.	<i>[Signature]</i>	20.12	Кафедра «ЕТЕМ»		
Затверд.		Муха А.М.	<i>[Signature]</i>	20.12	7.141.190034.03		

НАПИСАННЯ КОДУ В Arduino IDE

```

1 // Підключення аналогової та цифрової частини
2 const int redPin = 9;
3 const int greenPin = 10;
4 const int bluePin = 11;
5 const int potPin = A0;
6
7 void setup() {
8     // Підключення аналогової частини до Arduino IDE
9     pinMode(redPin, OUTPUT);
10    pinMode(greenPin, OUTPUT);
11    pinMode(bluePin, OUTPUT);
12 }
13
14 void loop() {
15     // Зчитування значення потенціометра
16     int potValue = analogRead(potPin);
17
18     // Зчитування значення аналогового входу до Arduino IDE
19     int redValue = map(potValue, 0, 1023, 255, 0);
20     int greenValue = map(potValue, 0, 1023, 0, 255);
21     int blueValue = map(potValue, 0, 1023, 0, 255);
22
23     digitalWrite(redPin, 255);
24     digitalWrite(greenPin, 0);
25     digitalWrite(bluePin, 255);
26     delay(100);
27     digitalWrite(redPin, 0);
28     digitalWrite(greenPin, 255);
29     digitalWrite(bluePin, 0);
30     delay(100);
31     digitalWrite(redPin, 255);
32     digitalWrite(greenPin, 0);
33     digitalWrite(bluePin, 0);
34     delay(100);
35 }

```

Вигляд коду на блок-схему напруги пілот сигналу

```

4 // Підключення аналогової частини до Arduino IDE
5 const int redPin = 9;
6 const int greenPin = 10;
7 const int bluePin = 11;
8 const int potPin = A0;
9
10 void setup() {
11     // Підключення аналогової частини до Arduino IDE
12     pinMode(redPin, OUTPUT);
13     pinMode(greenPin, OUTPUT);
14     pinMode(bluePin, OUTPUT);
15 }
16
17 void loop() {
18     // Зчитування значення потенціометра
19     int potValue = analogRead(potPin);
20
21     // Зчитування значення аналогового входу до Arduino IDE
22     int redValue = map(potValue, 0, 1023, 255, 0);
23     int greenValue = map(potValue, 0, 1023, 0, 255);
24     int blueValue = map(potValue, 0, 1023, 0, 255);
25
26     digitalWrite(redPin, 255);
27     digitalWrite(greenPin, 0);
28     digitalWrite(bluePin, 255);
29     delay(100);
30     digitalWrite(redPin, 0);
31     digitalWrite(greenPin, 255);
32     digitalWrite(bluePin, 0);
33     delay(100);
34     digitalWrite(redPin, 255);
35     digitalWrite(greenPin, 0);
36     digitalWrite(bluePin, 0);
37     delay(100);
38 }

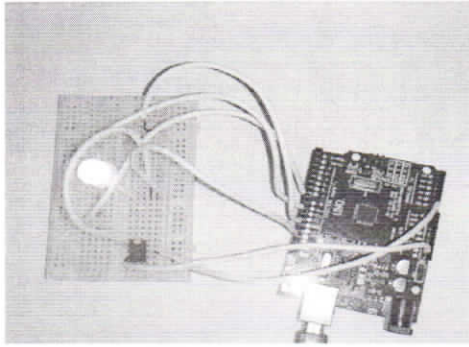
```

Також був написаний код для блок-схеми значення струмової установки

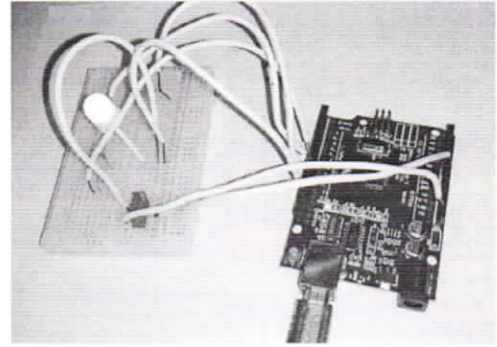
					Написання коду в Arduino IDE			
					Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля			
Змн.	Арк.	№ докум.	Підпис	Дата	Літ.	Маса	Масштаб	
Розроб.		Васильковський	<i>[Signature]</i>	20.12		1	1 : 1	
Перевір.		Муха.А.М.	<i>[Signature]</i>	20.12				
Т. Контр.					Арк.	77	Аркушів	79
Реценз.					МОН України. УДУНТ			
Н. Контр.		Карзова О.О.	<i>[Signature]</i>	20.12	Кафедра «ЕТЕМ»			
Затверд.		Муха А.М.	<i>[Signature]</i>	20.12	7.141.190034.04			

ЗАВАНТАЖЕННЯ КОДУ НА ПЛАТУ Arduino UNO

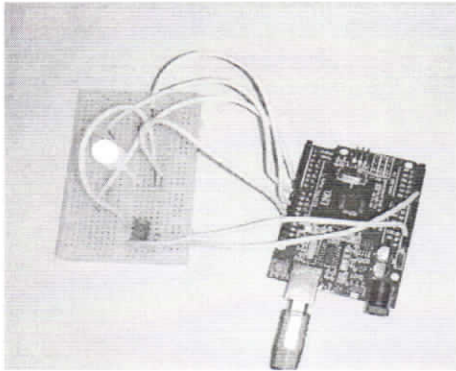
Робота RGB світло діоду відповідно написання коду, а саме напруги пілот сигналу



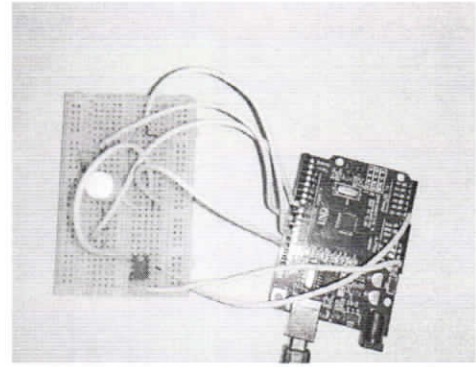
Подача 5В



Подача 3,75В

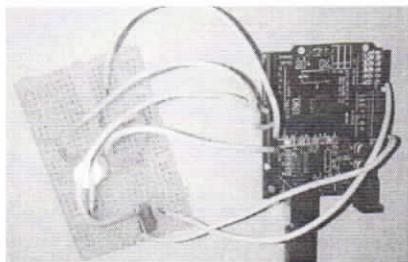


Подача 2,5В

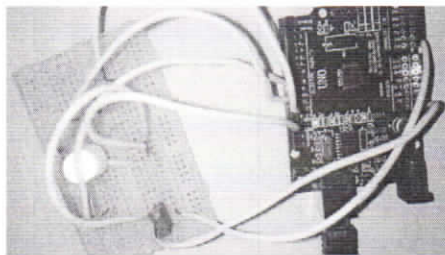


Подача нижче 2,5В

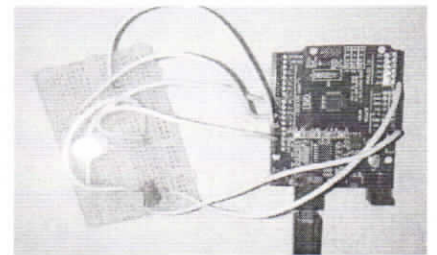
Робота RGB світло діоду відповідно до другого написання коду, а саме значення струмової установки



Подача 4,5В



Подача 4В



Подача 3,5

Завантаження коду на плату Arduino UNO								
						Ліг.	Маса	Масштаб
Змн.	Арк.	№ докум.	Підпис	Дата	Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля		1	1 : 1
		Васильковський	<i>[Signature]</i>	20.12				
		Муха.А.М.	<i>[Signature]</i>	20.12				
						Арк. 78	Аркушів 79	
						МОН України. УДУНТ		
		Карзова О.О.	<i>[Signature]</i>	20.12	7.141.190034.05	Кафедра «ЕТЕМ»		
		Муха А.М.	<i>[Signature]</i>	20.12				

ЗАМІРИ НАПРУГИ НА ПЛАТІ

Заміри для напруги пілот сигналу



5В білий



3,75 синій



2,52 зелений

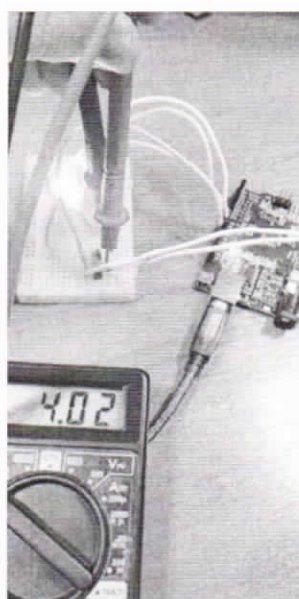


1,59 червоний

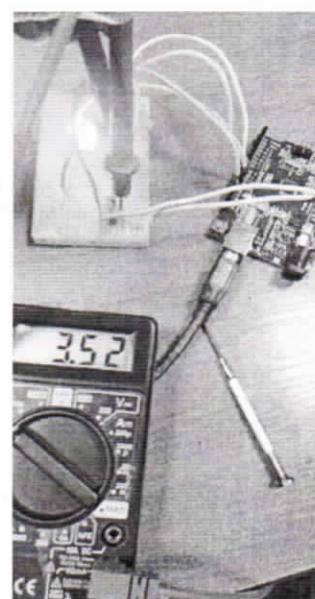
Заміри напруги для значення струмової установки



4,51В фіолетовий



4,02В помаранчевий



3,52В бірюзовий

					Заміри напруги на платі			
Змн.	Арк.	№ докум.	Підпис	Дата	Дослідження перехідних процесів у колах керування зарядного комплексу електромобіля 7.141.190034.06	Літ.	Маса	Масштаб
							1	1 : 1
Розроб.		Васильковський	<i>[Signature]</i>	20.12				
Перевір.		Муха.А.М.	<i>[Signature]</i>	11/12		Арк.	79	Аркушів 79
Т. Контр.						МОН України. УДУНТ		
Реценз.					Кафедра «ЕТЕМ»			
Н. Контр.		Карзова О.О.	<i>[Signature]</i>	21/12				
Затверд.		Муха А.М.	<i>[Signature]</i>	21/12				