

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)
(рівень вищої освіти)

Голографічний проектор на базі платформи Arduino Nano

Виконав: студент 2 курсу, групи СКСм-20-1

Шапошник К.А.
(прізвище, ініціали)

Спеціальність 123 Комп'ютерна інженерія

Тип програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Спеціалізовані комп'ютерні системи

Спеціалізовані комп'ютерні системи
(повна назва освітньої програми)

Керівник роботи док. Чумаченко С.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Чумаченко С.В.
(прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління

Кафедра Автоматизації проектування обчислювальної техніки

Рівень вищої освіти другий (магістерський)

Спеціальність 123 Комп'ютерна інженерія
(шифр і назва)

Тип програми Освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Освітня програма Спеціалізовані комп'ютерні системи
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« 4 » 11 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Шапошнику Костянтину Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи (проекту) Голографічний проектор на базі платформи Arduino Nano

Holographic projector based on the Arduino Nano platform

затверджена наказом по університету від « 04 » 11 2021 р. № 1635 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20.12.2021

3. Вихідні дані до роботи (проекту) _____

додаток Pixel

мова програмування Arduino

модулі та платформа Arduino

середовище розробки Arduino IDE

4. Перелік питань, що потрібно опрацювати у роботі _____

аналіз принципу роботи приладу

дослідження існуючих аналогів на ринку

проектування технології голографу

проектування системи управління приладом

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 13 слайдів

6. Консультанти розділів роботи (проекту)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

7. Дата видачі завдання 02.09.2021

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи (проекту)	Термін виконання етапів роботи	Примітка
		02.09.2021-08.09.2021	
2	Аналіз принципу роботи приладу	09.09. 2021-15.09. 2021	
3	Дослідження існуючих аналогів на ринку	16.09.2021-29.09.2019	
4	Проектування технології	30.09. 2021-13.10.2021	
5	Аналіз комплектуючих та їх придбання	14.10. 2021-31.10. 2021	
6	Розробка скетчу для платформи Arduino	01.11. 2021-15.11. 2021	
7	Збірка тестового приладу та прошивка платформи	15.11. 2021-30.11. 2021	
8	Оформлення пояснювальної записки	01.12. 2021-15.12. 2021	
9	Захист проекту	20.12. 2021-28.12. 2021	

Студент _____
(підпис)

Керівник роботи (проекту) _____ проф. Чумаченко С.В. _____
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка містить 69 сторінок, 32 рисунок, 7 лістингів та 15 джерел за переліком посилань.

МІКРОКОНТРОЛЕР, ARDUINO NANO, ATMEGA328P, ДАТЧИК, ПРОЕКТОР, СКЕТЧ, ПРОГРАМАТОР, ПЕРИФЕРІЯ, МОДУЛЬ ХОЛЛА, АДРЕСНА СВІТЛОДІОДНА ЛЕНТА, АЛГОРИТМ

Об'єкт дослідження - голографічний проектор на базі платформи Arduino Nano.

Предмет дослідження – проектування голографічного проектора на базі платформи Arduino Nano.

Мета кваліфікаційної роботи – розробка та проектування голографічного проектора на основі платформи Arduino моделі Nano та адресної світлодіодної стрічки WS2812.

В ході виконання кваліфікаційної роботи був спроектований та розроблений спеціалізований голографічний проектор на базі популярної плати фірми Arduino моделі Nano, що базується на мікроконтролері ATmega328P. В розробленому пристрої були також задіяні периферійні модулі, такі як: датчик Холла для зчитування кількості оборотів, адресна світлодіодна стрічка WS2812 IP20 та моторчик на 12В. Мовою С було розроблено алгоритм для маніпулювання періодичною зміною кольорів на адресній світлодіодній стрічці.

ABSTRACT

The explanatory note contains 69 pages, 32 figures, 7 listings and 15 references.

MICROCONTROLLER, ARDUINO NANO, ATMEGA328P, SENSOR, PROJECTOR, SKETCH, PROGRAMMER, PERIPHERALS, HALL MODULE, ADDRESSED LED STRIP, ALGOR

The object of research is a holographic projector based on the Arduino Nano platform.

The subject of research is the design of a holographic projector based on the Arduino Nano platform.

The purpose of the qualification work is to develop and design a holographic projector based on the Arduino platform of the Nano model and the WS2812 address LED strip.

During the qualification work, a specialized holographic projector based on the popular Arduino board of the Nano model based on the ATmega328P microcontroller was designed and developed. Peripheral modules were also used in the developed device, such as: Hall sensor for reading the number of revolutions, address LED strip WS2812 IP20 and 12V motor. An algorithm for manipulating the periodic change of colors on the address LED strip was developed in C language.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

АЦП - аналогово цифровий перетворювач

ШІМ - широтно-імпульсна модуляція

USB (Universal Serial Bus) - послідовний інтерфейс для підключення периферійних пристроїв до обчислювальної техніки.

IDE (Integrated development environment) - комплекс програмних засобів, який використовується програмістами для розробки програмного забезпечення

RGB (red, green, blue) - адитивна кольорова модель, що описує спосіб кодування кольору для відтворення кольору за допомогою трьох кольорів, які прийнято називати основними. Вибір основних кольорів обумовлений особливостями фізіології сприйняття кольору сітківкою ока.

GND (ground) - вузол ланцюга, потенціал якого умовно приймається за нуль, і вся напруга в системі відраховується від потенціалу цього вузла.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ.....	6
ВСТУП.....	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ	10
1.1 Сфера використання голографічних проєкторів.....	10
1.2 Дослідження принципу роботи типових голографічних проєкторів.....	11
1.3 Дослідження будови голографічних проєкторів.....	13
1.4 Оптичні ілюзії.....	14
1.5 Особливості людського зору.....	22
2 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ.....	27
2.1 Платформи Arduino.....	27
2.2 Порівняння існуючих платформ Arduino.....	31
2.3 Адресна світлодіодна стрічка.....	36
2.4 Мікро-двигун.....	41
2.5 Модуль датчика Холла.....	47
2.6 Bluetooth модуль.....	48
3 ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ ПРОЕКТУ	51
3.1 Середовище програмування Arduino IDE.....	51
3.2 Програмна реалізація системи управління.....	52
3.3 Додаток Pixel.....	60
3.4 Апаратні рішення та підключення установки.....	62
ВИСНОВКИ.....	66
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	67

ВСТУП

Сучасна технологія отримання тривимірних зображень залучила рекламодавців, особливо просування товарів та послуг. Така картинка приверне увагу та залишиться у підсвідомості. Є можливість керувати демонстраціями віддалено, перебуваючи у другій точці світу.

Голографічний проектор - це технічна новинка, яка виступає як унікальний і ефективний рекламний інструмент. Пристрій являє собою невелику конструкцію, вагою від 1 до 3 кг, і функціонує за принципом вентилятора. Обертові лопасті відображають проекцію будь-якого мультимедійного контенту. Це можуть бути зображення або анімаційні файли абсолютно будь-якого змісту.

Принцип роботи типових голографічних проекторів заснований на оптичному принципі *persistence of vision*. *Persistence of vision* традиційно відноситься до оптичної ілюзії, яка виникає, коли зорове сприйняття об'єкта не припиняється протягом деякого часу після того, як промені світла, що виходять від нього, перестають потрапляти в око, а також може означати те саме, що і «злиття-мерехтіння» - ефект, який полягає в тому, що зір здається постійним, коли світло, що потрапляє в очі, переривається з короткими та регулярними інтервалами

Arduino Nano входить в трійку найпопулярніших плат. Вона дозволяє створювати компактні пристрої, що використовують той же контролер, що і в Arduino Uno (ATmega328). Назва плати Nano говорить сама за себе - вона дійсно має невеликі розміри при тій же функціональності.

Свою популярність плати Arduino здобули через простоту у використанні. У вільному доступі можна здобути різноманітні датчики та модулі, які не потребують складного підключення до плати.

Arduino - торгова марка апаратно-програмних засобів для побудови простих систем автоматизації і робототехніки, орієнтована на непрофесійних користувачів. Як бренд була зароджена в Італії, проте через високу оригінальну

собівартість отримала популярність в китайських версіях-підробках. Користуючись китайськими інтернет-магазинами та сервісами доставки, можна замовити бажану плату або модулі для неї за низькою ціною, а велика кількість уроків та документації дозволить отримати достатній рівень знань.

У повсякденному житті люди найчастіше не звертають уваги на нескінченні ряди рекламних листівок з нав'язливою та нецікавою рекламою, особливо коли вона втрачає свою актуальність. Тому рекламні компанії повинні більше креативити та шукати способи звернення уваги на їх продукти з метою залучення клієнтів.

Оригінальним рішенням тут постають голографічні вентилятори. Використання таких пристроїв підвищує попит на рекламований продукт, оскільки привертає увагу пересічного прохожого.

На українському інтернет просторі голографічний вентилятор можна знайти за ціною від самих дешевих за 1400 гривень, до більш якісних за 5000 та вище.

Функціонал Arduino дозволяє навіть початківцю розробити більш дешевий аналог рекламного пристрою без втрати якості.

Метою даної роботи являється розробка та проектування голографічного проектора на основі платформи Arduino моделі Nano та адресної світлодіодної стрічки WS2812 як більш дешевого порівняно з професійними рекламними пристроями.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Сфера використання голографічних проекторів

Голографічний 3D-вентилятор –технологія, призначена для проєкції тривимірного зображення, яке «парить» у повітрі. Створювана картинка вражає високою чіткістю, насиченими кольорами, глибиною кольорів. Раніше створення такого зображення здавалося просто нереальною ідеєю, адже для його втілення потрібно дороге обладнання. Додатково необхідно було проводити монтаж важких конструкцій, що вимагало великих часових витрат. Завдяки новому девайсу може бути легко досягти результату - він простий у роботі, має невеликі розміри, споживає економну кількість електроенергії.

3D-вентилятор успішно замінив звичні банери, на які покупець часто не звертає уваги. Звичайна розтяжка з пропозицією про купівлю товару не порівнянна по ефективності з промальованою і продуманою до дрібниць голограмою. Повз неї просто неможливо пройти і не звернути увагу. Багато голографічних проекторів для реклами сприймаються як технології майбутнього, здатні зробити будь-який бренд таким, що запам'ятовується.

Сфера застосування голографічної технології широка. Вентилятори можна встановлювати в торгових центрах, автосалонах, метро, аеропортах, бутіках, метро, готелях, інших місцях із підвищеною прохідністю. Це не просто інформаційний екран для відтворення контенту, а інструмент для рекламного сегмента b2b.

Сучасна технологія отримання тривимірних зображень залучила рекламодавців, особливо просування товарів та послуг. Завдяки тривимірним зображенням створюється додатковий ажіотаж та інтерес до рекламованої продукції. Така картинка приверне увагу та залишиться у підсвідомості. Є можливість керувати демонстраціями віддалено, перебуваючи у другій точці світу.

Популярність застосування голографічних 3D-вентиляторів обумовлена рядом переваг.

1. Мобільність. Завдяки невеликій вазі та компактним габаритним розмірам його можна легко транспортувати з одного місця до іншого. Голограф легко кріпиться на стіні, а також можливе встановлення на підлозі на штативі або на вулиці під захисним ковпаком.

2. Зручність використання. Щоб здійснити підключення і почати працювати з обладнанням, не потрібно мати спеціальні навички, вміння.

3. WOW-ефект. Аксесуар привертає увагу, діє як візуальний магніт. Якщо зробити синхронізацію контенту з кількох 3D-вентиляторів, то вийде справжнє шоу, яке може привернути увагу цілої групи людей.

4. Велике охоплення аудиторії. Можна використовувати на конференціях, презентаціях, виставках, вечірках та інших сферах.

5. Можливість створення об'єднаної мережі з кількох вентиляторів із дистанційним керуванням. Чіткі, контрастні зображення, які чудово проглядаються з далекої відстані.

6. Широкі можливості. Можливість створити голографічну рекламу будь-якого товару, навіть, якого поки що немає.

7. Вигідна ціна поряд із хорошими якісними характеристиками.

1.2 Дослідження принципу роботи типових голографічних проекторів

Голографічний вентилятор — це пристрій для виведення зображення, що «левітує», з голографічним ефектом. Тобто. у такого проектора по суті немає заднього фону, а точніше його не видно для людського зору через особливості нашого сприйняття об'єктів, що швидко рухаються. На цьому і побудовано технологію безперервного руху - POV ефекту.

Сам же голографічний вентилятор, якщо говорити простою мовою - це по

суті звичайний вентилятор на лопатях якого встановлені LED пікселі високої щільності, які то спалахують, то тухнуть у певний момент часу, під час обертання. Все це створює POV ефект - який наше око сприймає, як об'ємну левіруючу голографічну проекцію об'єкта в повітрі.

Persistence of vision традиційно відноситься до оптичної ілюзії, яка виникає, коли зорове сприйняття об'єкта не припиняється протягом деякого часу після того, як промені світла, що виходять від нього, перестають потрапляти в око, а також може означати те саме, що і «злиття-мерехтіння» - ефект, який полягає в тому, що зір здається постійним, коли світло, що потрапляє в очі, переривається з короткими та регулярними інтервалами[1].

З моменту появи, термін PoV вважався поясненням сприйняття руху в оптичних іграшках, таких як фенакїстископ і зоотроп, а пізніше і в кіно. Проте ця теорія заперечувалась ще до появи кіно в 1895 році. Якщо "Persistence of vision" пояснити як "злиття-мерехтіння", його можна розглядати як фактор ілюзії зображень, що рухаються, в кіно і пов'язаних з ним оптичних іграшках, але не як його єдиний принцип. Найбільш важливим фактором у цьому типі анімації є стробоскопічний ефект, докладно описаний винахідником Саймоном Штампфером, але теоретики кіно принаймні за межами Німеччини та Австрії ігнорують його. Ранні описи ілюзії часто приписували ефект виключно недосконалості ока, особливо сітківки.

3D-голографічний проектор має вбудований мікропроцесор, за допомогою якого надсилається певна кількість сигналів через проміжок часу кожному світлодіоду. Надшвидке миготіння світлодіодів згладжується оком людини, це дозволяє сприймати зображення як повноцінну мальовничу картинку. Сучасні високотехнологічні розробки роблять картинку настільки реалістичною, що виникає бажання до неї доторкнутися. Цей процес відбувається за принципом обертового колеса велосипеда. На тонких спицях розташовані світлодіоди, які відтворюють запрограмоване зображення.

Обладнання голографічного 3D-проектора зовні дуже схоже на вентилятор, але функціонал у нього інший. Прилад складається з платформи

управління та проєкційного блоку, від якого залежить якісне відтворення та формування об'ємних зображень. LED-світлодіоди щільно посаджені на блок проєкції з чотирьох сторін. Вони спалахують у запланованому порядку, створюючи феномен інерції зорового сприйняття.

Термін «POV-дисплей» використовувався для світлодіодних пристроїв, які складають зображення, відображаючи одну просторову частину за раз у швидкій послідовності, наприклад, один стовпець пікселів кожні кілька мілісекунд. Двовимірне відображення POV часто досягається за допомогою швидкого переміщення одного ряду світлодіодів лінійним або круговим шляхом. Ефект у тому, що зображення сприймається глядачем як єдине ціле, доки весь шлях завершується під час візуального контакту. Додатковий ефект часто полягає у створенні ілюзії левітуючого в повітрі зображення. Тривимірний POV-дисплей часто будується за допомогою двовірної сітки світлодіодів, яка переміщується або обертається в площині. Пристрої відображення POV можуть використовуватися у поєднанні із тривалими витримками камери для отримання ідеального ефекту. Типовий приклад цього можна побачити у використанні ліхтарів на велосипедах, які створюють візерунки.

1.3 Дослідження будови голографічних проєкторів

Голографічний проєктор - це технічна новинка, яка виступає як унікальний і ефективний рекламний інструмент. Пристрій являє собою невелику конструкцію, вагою від 1 до 3 кг, і функціонує за принципом вентилятора. Обертіві лопасті відображають проєкцію будь-якого мультимедійного контенту. Це можуть бути зображення або анімаційні файли абсолютно будь-якого змісту.

Рисунок 1.1 – Професійний голографічний дисплей

Більш дорогі представники таких проекторів можуть похвалитися високим розширенням екрану, тихим та потужним двигуном, високою швидкістю обертання стрічки та відтворенням контенту в різних форматах.

Вентилятор для голограм працює за принципом блискавичного обертання лопатей, відображаючи об'ємний файл зі знімного носія в повітрі. Бажаного результату досягають за рахунок світлодіодів, розташованих на лопатях.

Основа пристрою - процесор, функція якого полягає в своєчасній подачі сигналів на світлодіоди. При розробці професійних голографічних проекторів враховувалася будова людського ока: швидкість обертання лопатей і проекція зображення при взаємодії сприймаються для людини як плаваюча в повітрі картинка або анімація[2].

Принцип дії кожного голографічного вентилятора базується на обертанні світлодіодної стрічки або панелі за допомогою колекторних або безколекторних моторів. Такі двигуни можуть живитися від літєвих акумуляторів або від мережі 220В через блоки живлення або регулятори.

Голографічний вентилятор є пристроєм, що складається з тонких лопатей, в які вбудовано безліч яскравих світлодіодів RGB високої щільності. Під час роботи голографічного 3D вентилятора на світлодіоди подаються електричні імпульси, синхронізовані з частотою обертання лопатей, у результаті формується яскрава картинка.

Лопаті дуже тонкі, а частота обертання дуже висока. Завдяки цьому ефект обертання є непомітним візуально. За рахунок постійного переміщення світлодіодів із великою швидкістю забезпечується 3D ефект. Тому пристрій фактично працює як голографічний проектор, створюючи яскраве тривимірне зображення, яке буквально повисає у повітрі. Технологія дозволяє виводити динамічну картинку будь-якого рівня складності.

1.4 Оптичні ілюзії

Оптична ілюзія, також зорова ілюзія - помилка в зоровому сприйнятті, викликана неточністю або неадекватністю процесів неусвідомлюваної корекції зорового образу, наприклад: місячна ілюзія, неправильна оцінка довжини відрізків, величини кутів або кольору зображеного об'єкта, а також фізичними причинами: «сплюснутий Місяць», «зламана ложка» у склянці з водою. Причини оптичних ілюзій досліджують як із розгляду фізіології зору, і у межах вивчення психології зорового сприйняття.

У художніх зображеннях навмисне спотворення перспективи викликає особливі ефекти, найкраще відомі з робіт Моріса Ешера. Створення оптичних ілюзій часто було темою робіт Сальвадора Далі, наприклад, його картини «Невільничий ринок із явищем незримого погруддя Вольтера», «Лебеді, що відбиваються у слонах» (рис. 1.2).

Рисунок 1.2 - «Невільничий ринок із явищем незримого погруддя Вольтера» та «Лебеді, що відбиваються у слонах» Сальвадора Далі

Існує ефект зорового сприйняття, коли спостерігач свідомо або мимоволі дає не відповідне дійсності пояснення картині, що спостерігається ним. Він добре відомий кожному, хто спостерігав хмари, що біжать по небу, часом приймають форму відомих зорових об'єктів. Такий же ефект може мати місце при спостереженні картини розподілу тіней тривимірних об'єктів при деяких ракурсах по відношенню до джерел світла. Саме цим пояснюються повідомлення про спостереження на Марсі людської голови, що мають форму сенсації в засобах масової інформації тощо.

Водночас пояснення випадково створеної плями іноді використовується

психологами для з'ясування властивостей інтелекту випробуваного, у якого обман зору викликається навмисно.

Вже близько ста років відомо, що коли на сітківці ока виникає зображення, що складається зі світлих і темних областей, світло від яскраво освітлених ділянок ніби перетікає на темні ділянки. Це називається оптичною іррадіацією.

Одна з таких ілюзій описана в 1995 професором Массачусетського технологічного інституту Едвардом Адельсоном в доповіді «Ілюзія тіні Адельсона». Він звернув увагу, що сприйняття кольору суттєво залежить від фону та однакові кольори на різному фоні сприймаються нами як різні, навіть якщо знаходяться близько та видно нами одночасно.

Ілюзії сприйняття глибини - неадекватне відображення сприйманого предмета та його властивостей. В даний час найбільш вивченими є ілюзорні ефекти, що спостерігаються при зоровому сприйнятті двовимірних контурних зображень. Мозок несвідомо бачить малюнки лише одноопуклі. Сприйняття залежить від напрямку зовнішнього, реального або освітлення, що передбачається.

На початку ХХІ століття інтерес до зорових спотворень продовжує зростати — з'являються нові наукові теорії, за допомогою яких вчені намагаються пояснити механізми виникнення оптичних ілюзій. Відповідно до однієї з них, спотворення відбуваються через те, що людський мозок постійно «передбачає» зображення, щоб компенсувати затримку між самою подією та моментом її сприйняття. Нейробіолог Марк Чангізі вважає, що саме спробами мозку "передбачити" картинку пояснюються деякі зорові спотворення.

Експерименти Чангізі та його колег із Каліфорнійського технологічного інституту показують, що цій теорії не суперечить жодна з класичних оптичних ілюзій. Серед найбільш показових прикладів «пророкування» зображення мозком Чангізі називає знамениту ілюзію Герінга (рис. 1.3). Коли людина рухається вперед, видимі ним об'єкти рухаються радіальними лініями, тому мозок схильний сприймати подібні зображення як ознака переміщення в

просторі. «Ці механізми чудово працюють у реальному житті, але ж вони змушують мозок помилятися, коли людина бачить радіальні лінії і при цьому залишається на місці», - зазначає дослідник.

Рисунок 1.3 - Ілюзія Геринга

Винахід магнітно-резонансної томографії стало справжнім подарунком для дослідників оптичних ілюзій - наука нарешті змогла хоча б загально зрозуміти, що відбувається в мозку людини при їх сприйнятті. Так, вивчаючи мозкову діяльність людини, що дивиться на куб Неккера, вчені зробили висновок, що мозок неоднозначно сприймає глибину зображення. Нейрони ніби «сперечаються» між собою, яку картинку слід вважати «істинною», в результаті спостерігач бачить куб то в одному положенні, то в іншому.

Завдяки сучасним методам досліджень людство знає, що за сприйняття відтінків кольору, форм предметів та їх переміщення у просторі відповідають різні ділянки мозку, але як ми отримуємо цілісне зображення, багато в чому залишається загадкою. Ентузіасти розробляють нові і нові способи обдурити зір, переосмислюючи і доповнюючи класичні ілюзії. Дивлячись на них, ми старанно «дозволяємо» власному мозку ввести нас в оману, а в результаті виникає більше запитань, ніж відповідей[3].

У наш час інтерес до проблеми такий високий, що протягом десяти років фахівці щороку проводять конкурс на кращу оптичну ілюзію. У 2014 році цю нагороду отримала динамічна ілюзія Еббінгауза, яка набагато переконливіше обманює зір, ніж класичний статичний варіант (рис. 1.4). За словами невролога Сюзанни Мартінес-Конде, яка входить до складу журі конкурсу, за рахунок постійної зміни розмірів суміжних об'єктів ефект нової ілюзії в кілька разів сильніший, ніж у нерухомого зображення, запропонованого Германом Еббінгаузом.

Рисунок 1.4 – Статична ілюзія Еббінгауза

Мартінес-Конде визнає, що більшість сучасних досліджень оптичних ілюзій ґрунтується на роботі, виконаній вченими XIX-го століття. Герман Гельмгольц першим зрозумів, що людські очі постійно роблять швидкі узгоджені рухи, звані саккади. Щоб зрозуміти про що мова, закрийте одне око і злегка натисніть пальцем на нижню повіку іншого - «картинка», яку бачить ваш мозок відразу ж почне рухатися. У звичайному житті ми не помічаємо цих мікроскопічних «підсмикувань», тому що мозок давно навчився згладжувати зображення, але коли він стикається з незвичною ситуацією, такою як механічна дія на очне яблуко, саккади виявляють себе.

На думку Сюзанни, саме саккади відіграють ключову роль у знаменитій ілюзії «Змії, що обертаються» (рис. 1.5), яку розробив японський психіатр Акіюші Китаока. У ході експериментів зі «Зміями» Мартінес-Конде та її колеги з'ясували, що при погляді на ілюзію активізуються ті ж нейрони, що і при погляді з вікна поїзда, що швидко рухається, коли здається, що пейзаж «їде повз», а не навпаки. При цьому якщо за допомогою деяких хитрощів змусити спостерігача припинити саккади, ілюзія зникає.

Рисунок 1.5 - Ілюзія «Змії, що обертаються»

Невролог пояснює це так: видимість руху в «Зміях, що обертаються» створюється за рахунок великої кількості оптичної інформації, що надходить у різні ділянки сітківки очей. Певне поєднання світлових сигналів обманює мозок, змушуючи його сприймати статичне зображення як динамічне. Саккади постійно оновлюють «картинку», не даючи мозку адаптуватися до неї, якщо

вони призупиняються, через деякий час - йде і ілюзія руху.

Як і багато інших фахівців з оптичних ілюзій, Сюзанна Мартінес-Конде впевнена — далеко не всі механізми зорового сприйняття відкриті, а ті, що вже відомі, поки що не дуже добре вивчені.

Одним із найцікавіших аспектів ілюзії є створення візуальної безперервності та руху за допомогою нерухомих зображень. Незважаючи на те, що технології, можливо, змінили спосіб анімації сьогодні, вони не змінили базовий механізм цього ремесла.

Постійність зору (persistence of vision) - це оптичне явище, у якому створюється ілюзія руху, оскільки мозок інтерпретує кілька нерухомих зображень як одне. Коли кілька зображень з'являються в досить швидкій послідовності, мозок змішує їх в одне постійне зображення, що рухається (рис. 1.6).

Людське око та мозок можуть обробляти лише близько 12 окремих зображень за секунду, зберігаючи зображення протягом 1/16 секунди. Якщо наступне зображення замінюється протягом цього періоду часу, створюється ілюзія безперервності.

Способи створення руху із PoV.

1. Анімація по двоє - коли одне зображення відображається на кожні два кадри із загальною швидкістю 12 кадрів на секунду, це забезпечує плавний рух.

2. Анімація по одному - коли для кожного кадру відображається одне зображення із загальною швидкістю 12 кадрів в секунду, це забезпечує ще швидший рух.

Рисунок 1.6 - Принцип створення анімації зі сталістю зору

У 19 столітті англо-швейцарський фізик Пітер Марк Роже описав сталість зору як тип очного дефекту, при якому об'єкти, що рухаються, виглядали нерухомо, коли вони досягли досить високої швидкості. Уявіть стельовий вентилятор. Ви не можете сказати, скільки у нього важелів, коли він обертається

з максимальною швидкістю, але коли він зупиняється, стає ясно, скільки їх.

Але вірне й протилежне. Перегляд окремих зображень на високій швидкості створює ілюзію руху. Винахід Джозеф Плато фенакистоскопа (рис. 1.7) доводить це. Якби окремі малюнки містилися на круговий диск та оберталися, здавалося б, що малюнки рухаються.

Рисунок 1.7 - Фенакистоскоп

Ці теоретики більше орієнтувалися на те, як око зберігає зображення. Однак було виявлено, що це явище пов'язане не стільки з тим, як довго мозок може зберегти зображення, скільки з тим, як мозок взаємодіє та обчислює всі збережені дані. Саме це постійне та складне обчислення створює сталість зору.

Пізніше, на початку 20-ого століття, німецькі теоретики, такі як Макс Ветхаймер і Хьюго Мюнстерберг, дійшли висновку, що цей процес насправді відбувається за сітківкою, а мозок є двигуном, що стоїть за ілюзією.

Око сприймає безліч різних відчуттів – колір, світло, глибину, форму та рух. Всі ці дані вирушають у різні області мозку. Саме цей постійний потік та зв'язок між різними шляхами та областями мозку дають нам унікальне сприйняття та здатність спостерігати такі явища[4].

1.5 Особливості людського зору

Людина бачить не очима, а за допомогою очей, звідки інформація передається через зоровий нерв, хіазму, зорові тракти у певні області потиличних часток кори мозку, де формується та картина зовнішнього світу, що ми бачимо. Всі ці органи і складають наш зоровий аналізатор або зорову систему.

Наявність двох очей дозволяє зробити наш зір стереоскопічним, тобто формувати тривимірне зображення. Права сторона сітківки кожного ока передає через зоровий нерв "праву частину" зображення в праву сторону головного мозку, аналогічно діє ліва сторона сітківки. Потім дві частини зображення –

праву та ліву – головний мозок поєднує воедино.

Так як кожне око сприймає «свою» картинку, при порушенні спільного руху правого та лівого очей може бути засмучений бінокулярний зір. Просто кажучи, у вас почне двоїтися в очах або ви одночасно бачитимете дві зовсім різні картинки.

Рисунок 1.8 – Будова людського ока

Наші очі еволюціонували протягом 500 мільйонів років, починаючи з простої світлочутливої точки. Це дуже важливий еволюційний крок, тому що зрячі істоти мають явні переваги перед сліпими особами. Дослідники ніяк не можуть дійти єдиної думки щодо того, чи відбулися ці очі від одного протоока або ж у кожному окремому випадку очі розвивалися незалежно один від одного. Потреби різних організмів призвели до виникнення різних типів очей: від плоских очей, очей у вигляді ямки, очей у вигляді отвору, а також фасеткових чи складних очей до очей із кришталиком у хребетних, включаючи людей. Останній тип очей вважається одним із найбільш досконалих органів зору, створених природою. Розвиток очей з кришталиком дозволив чітко і ясно сприймати довкілля. Незважаючи на це, людське око має свої еволюційні слабкості.

Працюючи разом із очима, мозок грає значної ролі у світі людського зору. Докладаючи непомітних мінімальних зусиль, він компенсує недоліки наших очей. Коли око з кришталиком у хребетних еволюціонувало, сталося щось незвичайне. Наприклад, на відміну від каракатиці, що має дуже складне міхуроподібне око з кришталиком, що з'явилося в результаті вп'ячування епідермісу, людське око формувалося зовсім по-іншому в міру розвитку мозку. На перший погляд, це може здатися незначною відмінністю, до того ж найчастіше кращою є наявність більшої кількості зорових рецепторів. Як це не дивно, наші зорові рецептори розташовані неправильно навколо сітківки і спрямовані всередину організму, а нервові клітини – до джерела світла. Це

означає, що у нас «перевернуте око», через що мозку потрібно встановлювати предмети відповідно до правильної перспективи. Це також означає, що люди та інші хребетні мають так звану сліпу пляму.

Сліпа пляма - місце в очах, в якому зоровий нерв проходить через сітківку в мозок. Мережа нервових клітин, що є частиною зорового нерва, створює таку собі подобу «отвори» в сітківці, тобто. частина поля зору, яка не сприймається через нестачу світлочутливих зорових фоторецепторів. Ця невдала конструкція сітківки, що призводить до виникнення сліпої плями в нашому полі зору, називається у фахівців «перевернутим оком». Сліпа пляма розташована на 15 градусах до носа щодо зорової лінії. Здорові люди не помічають недостатньої зорової інформації, оскільки наш мозок інтерполює сліпу пляму на підставі інформації про навколишнє середовище, інформації від іншого ока та обробки зображень, що отримуються в результаті рухів очей. Сліпа пляма була вперше описана французьким фізиком Едмом Маріоттом в 1660 році (рис. 1.9).

Рисунок 1.9 – Демонстрація «сліпої плями»

Так само, як і сліпа пляма, кожне людське око має зону на сітківці, в якій забезпечується максимально сфокусований зір – так звана макула. Центр макули містить найбільшу концентрацію колб, одного з двох типів зорових фоторецепторів ока. Ця невелика центральна ямка розташовується точно в середині макули і відповідає за чіткий центральний зір.

Можна сказати, що всі істоти мають очі, які вони заслуговують. Потенційним жертвам хижаків важливо мати чудове кругове поле зору. Саме тому очі у зайців, оленів та інших травоядних тварин очі розташовані на боках голови. Однак це ускладнює сприйняття глибини та відстані.

Завдяки очам, спрямованим уперед, люди можуть дуже точно оцінювати глибину та відстань, хоча в нас і немає кругового поля зору, можливо через те, що вона нам не потрібна.

Строго кажучи, ми не можемо пильно дивитися на предмет, коли фокусуємось на ньому. Зорові фоторецептори на нашій сітківці реагують лише на зміни умов освітлення. Таким чином, якщо ми почнемо дивитися на предмет, нерухоме зображення почне розпливатися. Природа, як завжди, має рішення: наші очі безперервно роблять невеликі рухи, чого ми навіть не помічаємо, щоб предмет був у фокусі і ми могли сприймати предмети навколо себе. Навіть коли ми фокусуємо погляд в одній точці, наші очі роблять невеликі та швидкі переміщення, відомі як саккада.

Периферійний зір є частиною зору, що знаходиться поза центральним сфокусованим полем зору. Метою периферійного зору є формування початкового уявлення чи контексту перед тим, як ви зможете сфокусуватися на чомусь, тому воно працює не так, як сфокусований зір. Периферійний зір займає більше 90% поля зору, використовуючи близько 50% зорових фоторецепторів. Це означає, що в нашому периферійному зорі здатність розрізняти дрібні деталі відходить на інший план через значно меншу гостроту зору чи дозволу. Проте наш периферійний зір набагато краще дозволяє сприймати рух, бо нам потрібно швидко розпізнавати потенційні ризики[5].

2 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

2.1 Платформи Arduino

Arduino - це електронний конструктор та зручна платформа швидкої розробки електронних пристроїв для новачків та професіоналів. Платформа користується величезною популярністю у всьому світі завдяки зручності та простоті мови програмування, а також відкритій архітектурі та програмному коду. Пристрій програмується за допомогою USB без використання програматорів.

Перший прототип контролера був випущений ще 2005 року, коли Массімо Банці розробив його для студентів Інституту проектування взаємодій міста Івреа, Італія. Назва пристрою походить від імені короля Ардуїна, який правив Італією всього два роки на початку XI століття, на честь якого був названий і пивний бар "ді Ре Ардуїно", що належить Массімо Банці, і розташований на тому самому місці, де за переказами народився король Ардуїн.

Arduino дозволяє комп'ютеру вийти за рамки віртуального світу у фізичний та взаємодіяти з ним. Пристрої на базі Arduino можуть отримувати інформацію про навколишнє середовище за допомогою різних датчиків, а також керувати різними виконавчими пристроями.

Arduino - це ефективний засіб розробки програмованих електронних пристроїв, які, на відміну персональних комп'ютерів, орієнтовані тісно взаємодію з навколишнім світом. Arduino - це відкрита програмована апаратна платформа для роботи з різними фізичними об'єктами і є простою платою з мікроконтролером, а також спеціальне середовище розробки для написання програмного забезпечення мікроконтролера.

Arduino може використовуватися для розробки інтерактивних систем, що керуються різними датчиками та перемикачами. Такі системи, у свою чергу, можуть керувати роботою різних індикаторів, двигунів та інших пристроїв. Проекти Arduino можуть бути як самостійними, так і взаємодіяти з програмним

забезпеченням, що працює на персональному комп'ютері (наприклад, програмами Flash, Processing, MaxMSP). Будь-яку плату Arduino можна зібрати вручну або купити готовий пристрій; середовище розробки для програмування такої плати має відкритий вихідний код і є повністю безкоштовним.

Мікроконтролер на платі програмується за допомогою мови Arduino та середовища розробки Arduino (заснована на середовищі Processing). Мова програмування Arduino є реалізацією схожої апаратної платформи "Wiring", яка базується на середовищі програмування мультимедіа "Processing". Плати можуть бути зібрані користувачем самостійно або куплені у зборі. Стандартні бібліотеки Arduino відкривають простір для творчості у плані автоматизації будь-чого.

Програмування плати здійснюється через спеціальну програмну оболонку – IDE, яку можна завантажити безкоштовно на сайті Arduino. Написана на Java, доброзичлива оболонка працює під Windows, Mac OS і Linux, вона містить текстовий редактор, менеджер проектів, препроцесорний компілятор, а також інструменти для завантаження програми безпосередньо в мікроконтролер.

Мікроконтролери, що використовуються в Arduino, вже мають прошитий завантажувач (bootloader), тому програматор не потрібен, достатньо з'єднати плату з комп'ютером через USB або через перехідник UART-USB, і завантажити програму.

Є на платі і можливість прошити завантажувач у мікроконтролер самостійно за допомогою програматора, в Arduino IDE вбудована підтримка найпопулярніших дешевих програматорів, є штирьовий роз'єм для внутрішньосхемного програмування - ICSP для AVR, JTAG для ARM.

У більшості пристроїв Arduino використовуються мікроконтролери Atmel AVR ATmega328, ATmega168, ATmega2560, ATmega32U4, ATTiny85 із частотою тактування 16 або 8 МГц. Існують також плати на процесорі ARM Cortex M.

Arduino надає великі можливості для створення будь-яких пристроїв, можна підключати датчики, замки, мотори, дисплеї, роутери, навіть чайники. Можна розширювати виріб додатковими платами - шилдами, наприклад для роботи з GPS, для з'єднання по локальній мережі або інтернету, для bluetooth,

Wi-Fi і т.д.

Зручно те, що для підключення розширень не потрібен паяльник, використовуються прості штирьові з'єднання, що дозволяє легко конструювати макети, ускладнювати їх так, як хочеться, загалом, простір для творчості безмежний.

Платформа Arduino отримала широке визнання у розробників нових електронних пристроїв, викладачів та студентів інженерних напрямів підготовки, а також школярів у гуртках технічної творчості.

Використання Arduino полегшує процес роботи з мікроконтролерами. По технічному оснащенню вона ідеально підходить для освітнього процесу з проектування різних мехатронних систем і роботів, завдяки зрозумілому середовищу програмування та можливості спостереження фізичних процесів у реальному часі, а також завдяки зрозумілому середовищу програмування та ряду інших переваг.

Вона може використовуватися як засіб навчання та дослідження в цифровій обробці сигналів, електроніці, схемотехніці, робототехніці, автоматичі та інші. Більш потужні плати Arduino застосовні для вирішення складних технічних завдань, пов'язаних з розробкою великих проектів та їх комплексною автоматизацією.

Arduino – це найбільш популярна тенденція, що робить мікроконтролери доступними для розуміння та використання великої кількості людей, навіть не фахівців у цій галузі. За допомогою цієї популярної платформи можна зробити велику кількість цікавих та корисних проектів[6].

Можна сказати, що Arduino є універсальним програмованим контролером-конструктором, що розширюється, який може стати незамінним помічником при вирішенні будь-яких творчих завдань, пов'язаних з електронікою довільного призначення, хоч будильник, хоч складний робот, хоч кроковий двигун, - усім цим, і не тільки, можна керувати за потрібним алгоритмом за допомогою Arduino.

Величезна кількість різноманітної периферії: кнопки, датчики, світлодіоди, РК-індикатори та інші органи взаємодії з навколишнім світом,

доступні для роботи з Arduino.

В інтернеті вже доступні сотні програм для Arduino, здатні допомогти як початківцям, так і досвідченим користувачам для реалізації їхніх проєктів.

Існує безліч інших мікроконтролерів та мікропроцесорних пристроїв, призначених для програмування різних апаратних засобів: Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard та багато інших. Всі ці пристрої пропонують схожу функціональність і покликані звільнити користувача від необхідності заглиблюватися в дрібні деталі внутрішнього пристрою мікроконтролерів, надавши йому простий та зручний інтерфейс їх програмування. Arduino також спрощує процес роботи з мікроконтролерами, але на відміну від інших систем надає низку переваг для викладачів, студентів та радіоаматорів.

1. Низька вартість. Порівняно зі схожими апаратними платформами, плати Arduino мають відносно низьку вартість: готові модулі Arduino коштують не дорожче 50\$, а можливість зібрати плату вручну дозволяє максимально заощадити кошти та отримати Arduino за мінімальну ціну.

2. Кросплатформність. Програмне забезпечення Arduino працює на операційних системах Windows, Macintosh OSX і Linux, тоді як більшість подібних систем орієнтовані на роботу тільки в Windows.

3. Просте та зручне середовище програмування. Середовище програмування Arduino зрозуміле і просте для початківців, але при цьому досить гнучке для просунутих користувачів. Вона заснована на середовищі програмування Processing, що може бути зручним для викладачів. Завдяки цьому студенти, які вивчають програмування в середовищі Processing, зможуть легко освоїти Arduino.

4. Розширюване програмне забезпечення з відкритим кодом. Програмне забезпечення Arduino має відкритий вихідний код, завдяки цьому досвідчені програмісти можуть змінювати та доповнювати його. Можливості мови Arduino можна також розширювати за допомогою бібліотек C++. Завдяки тому, що він заснований мовою AVR C, просунуті користувачі, які бажають розібратися в технічних деталях, можуть легко перейти з мови Arduino на C або вставляти

ділянки AVR-C коду безпосередньо в Arduino.

5. Відкрите апаратне забезпечення, що розширюється. Пристрої Arduino побудовані на базі мікроконтролерів Atmel ATmega8 та ATmega168. Завдяки тому, що всі схеми модулів Arduino опубліковані під ліцензією Creative Commons, досвідчені інженери та розробники можуть створювати свої версії пристроїв на основі наявних. І навіть звичайні користувачі можуть збирати досвідчені зразки Arduino для кращого розуміння принципів їхньої роботи та економії коштів.

2.2 Порівняння існуючих платформ Arduino

Оригінальний Arduino був розроблений для одного специфічного завдання і впорався з цим завданням досконало. З успіхом першої оригінальної плати Arduino компанія вирішила створити більше проєктів, деякі з них для дуже специфічних завдань. Крім того, оскільки оригінальний дизайн Arduino був під відкритою ліцензією, кілька компаній та приватних осіб розробили свої власні Arduino сумісні плати розширень, або за принципами відкритого вихідного коду, запропонували свої зміни в Arduino. Arduino розпочав програму сертифікації для забезпечення сумісності з бордами, які використовують різні процесори, і Intel Galileo був першим, хто отримав подібний сертифікат. Будь-хто може зробити свій власний Arduino-сумісний прилад, але найменування та логотип Arduino зарезервовані як торгова марка. Таким чином, ви знайдете безліч плат з іменами, що закінчуються на "uino", що мають на увазі сумісність.

Arduino зробив дизайн плати з відкритим вихідним кодом, але вони, як і раніше, виробляють плати самостійно. Ці плати відомі як офіційні. Інші компанії також роблять Arduino сумісні плати.

Залежно від поставленої мети, можна підібрати найбільш підходящу платформу Arduino серед 22 офіційно представлених моделей. Кожна з яких має свої переваги і недоліки.

Arduino Uno є стандартною платою Arduino і, можливо, найпоширенішою. Вона заснована на чіпі ATmega328, що має на борту 32 КБ флеш-пам'яті, 2 КБ

SRAM та 1 Кбайт EEPROM пам'яті. На периферії має 14 дискретних (цифрових) каналів введення/виводу та 6 аналогових каналів вводу/виводу, це дуже різнобічно-корисні девайси, що дозволяють перекривати більшість аматорських завдань у галузі мікроконтролерної техніки. Дана плата контролера є однією з найдешевших і найчастіше використовуваних (рис. 2.3).

Рисунок 2.3 - Arduino Uno

Платформа Arduino Mini це крихітний пристрій, аналог Arduino Uno, що використовується в проектах, що вимагають максимальної економії місця. Вона містить 14 цифрових входів/виходів та 4 аналогових входних контакти. Ще чотири доступні, але не виведені. Пристрій настільки мініатюризований, що не має ні USB-роз'єму, ні регулятора потужності, ні навіть гребінки для підключення периферії. Програмування здійснюється за допомогою зовнішнього USB або RS232 через послідовний адаптер TTL (рис. 2.4).

Рисунок 2.4 - Arduino Mini

Контролер Arduino Pro заявлений у двох версіях, на основі ATmega168 та ATmega328. Версія 168 працює на 3,3 з тактовою частотою 8 МГц, а версія 328 працює на 5 В і частоті 16 МГц. Обидві версії мають 14 цифрових входів/виходів та 6 аналогових входів. Контролер має роз'єм живлення батареї JST, перемикач для вибору між режимами живлення, та простір, відведений для модуля живлення, при необхідності. На платі немає порту USB, але замість нього використовується кабель FTDI для програмування. Arduino Pro відрізняється від більшості інших Arduino тим, що будучи окремою самостійною макетною платою, він також може бути використаний для розширення функціоналу інших контролерів як шилд. Він випускається без портів та звичних гребінець. Усі цифрові та аналогові входи та виходи

розташовані по краях плати, зберігаючи стандартне для Arduino розташування отворів, готових до припаювання гребінок або дротів, за потребою. Замість використання для прототипування нових проектів Arduino Pro більше спрямований на остаточний монтаж у готовій продукції. Arduino Pro не розроблявся самим Arduino, а був розроблений та запущений у виробництво фірмою SparkFun Electronics (рис. 2.5)[8].

Рисунок 2.5 - Arduino Pro

Arduino Nano - одна з найпопулярніших плат. Це функціональний аналог Arduino Uno, але на мініатюрній платі. Відмінність полягає у відсутності власного гнізда для зовнішнього живлення, використання чіпа FTDI FT232RL для USB-Serial перетворення (або CH340G, потрібно встановити відповідні драйвера) та застосуванням mini-USB кабелю для взаємодії замість стандартного. В іншому, начинка та способи взаємодії збігаються з базовою моделлю. Платформа має штиркові контакти, що дозволяє легко встановлювати на макетну плату.

Arduino Nano — це один із типів плати мікроконтролера, розроблений Arduino.cc. Його можна створити за допомогою мікроконтролера, такого як Atmega328. Цей мікроконтролер також використовується в Arduino UNO. Це плата невеликого розміру, гнучка з широким спектром застосувань. Інші плати Arduino в основному включають Arduino Mega, Arduino Pro Mini, Arduino UNO, Arduino YUN, Arduino Lilypad, Arduino Leonardo і Arduino Due.

Ця плата має багато функцій, як і плата Arduino Duemilanove. Однак Nano відрізняється упаковкою. У ньому немає роз'єму постійного струму, щоб живлення можна було подавати за допомогою невеликого USB-порту, інакше безпосередньо під'єднувати до контактів, таких як VCC і GND. Ця плата може бути забезпечена напругою від 6 до 20 вольт за допомогою порту mini USB на платі.

Плата Arduino Nano схожа на плату Arduino UNO, включаючи аналогічний

мікроконтролер, наприклад Atmega328p. Таким чином, вони можуть працювати по подібній прошивці. Основна відмінність між цими двома — розмір, оскільки розмір Arduino Uno становить як розмір двох плат Nano. Таким чином, плати Uno займають більше місця в системі. Програмування UNO можна виконати за допомогою кабелю USB, тоді як Nano використовує кабель mini USB.

Зв'язок плати Arduino Nano може здійснюватися за допомогою різних джерел, наприклад, за допомогою додаткової плати Arduino, комп'ютера або мікроконтролерів. Мікроконтролер, що використовується в платі Nano (ATmega328), забезпечує послідовний зв'язок - UART TTL. Їм можна скористуватись через цифрові контакти, такі як TX та RX. Програмне забезпечення Arduino складається з послідовного монітора, що дозволяє легко передавати й отримувати текстову інформацію з плати.

Світлодіоди TX і RX на платі Nano будуть блимати щоразу, коли інформація надсилається через посилення FTDI та USB у напрямку до комп'ютера. Бібліотека SoftwareSerial забезпечує послідовний зв'язок на будь-якому з цифрових контактів на платі. Мікроконтролер також підтримує зв'язок SPI і I2C (TWI).

Програмування Arduino Nano можна виконати за допомогою програмного забезпечення Arduino. Мікроконтролер ATmega328 на платі Nano поставляється з попередньо запрограмованим завантажувачем. Цей завантажувач дозволяє завантажувати новий код без використання зовнішнього програмного забезпечення. Програмування можна здійснити за допомогою протоколу STK500. Тут також можна уникнути вбудованого завантажувача, а програму мікроконтролера можна завантажити за допомогою внутрішньосхемного послідовного програмування або ICSP з провайдером Arduino.

Всього на платі Nano є 14 цифрових і 8 аналогових контактів. Цифрові контакти можна використовувати для взаємодії з датчиками, використовуючи їх як вхідні контакти або керуючи навантаженнями, використовуючи їх як вихідні. Прості функції, такі як pinMode() і digitalWrite(), можна використовувати для керування їх роботою. Робоча напруга для цифрових контактів – від 0 В до 5 В. Аналогові контакти можуть вимірювати аналогову напругу від 0 В до 5 В за

допомогою будь-якого з 8 аналогових контактів, використовуючи просту функцію, наприклад `analogRead()`.

Ці контакти окрім того, що вони відповідають своєму призначенню, також можуть використовуватися для спеціальних цілей.

Послідовні контакти 0 (Rx) і 1 (Tx) використовуються для отримання та передачі послідовних даних TTL. Вони під'єднані до відповідного чіпа ATmega328P від USB до TTL.

Виводи зовнішнього переривання 2 і 3 можна налаштувати так, щоб ініціювати переривання за низьким значенням, наростаючим або спадним фронтом або зміною значення.

ШІМ-контакти 3, 5, 6, 9 і 11 забезпечують 8-розрядний вихід ШІМ за допомогою функції `analogWrite()`.

Контакти SPI 10 (SS), 11 (MOSI), 12 (MISO) і 13 (SCK) використовуються для зв'язку SPI.

I2C A4 (SDA) і A5 (SCA) використовується для зв'язку ІІС за допомогою бібліотеки `Wire`.

AREF - використовується для забезпечення опорної напруги для аналогових входів за допомогою функції `analogReference()`.

2.3 Адресна світлодіодна стрічка

Світлодіодна стрічка - джерело світла, зібране на основі світлодіодів. Є світлодіодним модулем на гнучкій друкованій платі, на якій рівновіддалено один від одного розташовані світлодіоди. Зазвичай ширина стрічки становить 8-20 мм, товщина зі світлодіодами 2-3 мм. При виготовленні стрічка змотується в рулони довжиною від 1 до 30 м. Для обмеження струму через світлодіоди в електричну схему стрічки вводяться баластові опори – резистори, які також монтуються на стрічці. Світлодіодні стрічки виготовляються з використанням SMD- та DIP-технологій. Цифри позначення означають розмір чіпа кристала в десятих частках міліметра[10].

Світлодіодні стрічки є новими і універсальними формами освітлення.

Існує багато варіантів і винятків, але здебільшого вони мають такі характеристики:

- складається з багатьох окремих світлодіодних випромінювачів, встановлених на вузькій гнучкій друкованій платі;
- працюють від низької напруги постійного струму;
- доступні в широкому діапазоні фіксованих і змінних кольорів і яскравості;
- поставляється в довгій котушці, її можна обрізати до потрібної довжини та включає двосторонній клей для кріплення (рис. 2.8).

Рисунок 2.8 – Звичайна світлодіодна стрічка

У конструкції RGB-стрічки використовуються розміщені на одній основі світлодіоди, що чергуються трьох кольорів - червоний, зелений, синій, тобто цю стрічку можна представити як три однокольорові стрічки, або трикомпонентні RGB-світлодіоди, що мають у своєму складі три напівпровідникових випромінювача світіння, об'єднані в одному корпусі.

Світлодіодна стрічка працює від постійного струму і підключається до джерела струму з постійною напругою величиною зазвичай 12, рідше 24 В і 5 В. Тому для підключення світлодіодної стрічки до мережі електроживлення додатково необхідний перетворюючий блок живлення.

Для плавного керування яскравістю та кольором світіння кольорової світлодіодної стрічки застосовуються контролери, принцип роботи яких полягає у зміні яскравості світіння світлодіодів окремо по кожному кольору. Багато контролерів можуть керуватися за допомогою пульта дистанційного керування.

Більшість стрічок мають обмеження по довжині послідовно підключених ділянок 5 метрів, тому реалізуючи проекти з великою кількістю стрічки слід використовувати паралельну схему підключення. При цьому також слід враховувати перетин дроту: чим більша відстань між блоком живлення і

стрічкою, тим вище втрати, і тим більше потрібно перетин дроту.

Розрахунок необхідної потужності блоку живлення здійснюється виходячи з номінальної потужності стрічки, довжини ділянок, що підключаються, а також коефіцієнта запасу, який зазвичай слід приймати як 1,15. Так для стрічки 240 SMD 3014, загальною довжиною в 4 метри буде потрібно блок живлення потужністю = 24 Вт (номінальна потужність стрічки) * 4 м * 1,15 (коефіцієнт запасу) = 110,4 Вт.

Переваги світлодіодних стрічок.

1. Простота монтажу. Багато стрічки мають на звороті клеючий шар, що дозволяє її легко кріпити практично на будь-які поверхні.

2. Невисока вартість експлуатації. Щодо світловий потік/вартість експлуатаційних витрат світлодіоди мають один із найвищих показників.

3. Надійність. У порівнянні з традиційними лампами розжарювання та люмінесцентними лампами, світлодіоди мають більший термін служби.

4. Необмежений потенціал збільшення світлового потоку в порівнянні з точковими джерелами, сумісними зі старою арматурою. Немає небезпеки перегріву елементів - світловий потік пропорційний довжині стрічки.

5. Можливість реалізації оригінальних дизайнерських рішень за рахунок гнучкості та невеликої товщини світлодіодної стрічки.

6. Можливість вибору бажаного колірної відтінку сцени при використанні RGB-світлодіодних стрічок з контролерами, що дозволяють керувати незалежно яскравістю кожного каналу і також застосування художніх ефектів підтримуваних більшістю контролерів: блимання, зміна кольорів, плавне перетікання відтінків з одного в інший, ефект «хвилі, що біжить», регулювання яскравості і т. д., як вручну, так і встановленими режимами роботи контролера.

– Відсутність потреби у додатковій вологозахисній арматурі при використанні вологозахисних світлодіодних стрічок та блоків живлення або розміщення блоків живлення поза приміщеннями з підвищеною вологістю.

– Електробезпека, зумовлена низькою напругою живлення самих стрічок за умови використання блоків живлення, що мають трансформатори та

гальванічну розв'язку між ланцюгами живлення стрічки та живлення самого блоку.

На відміну від звичайної світлодіодної RGB-стрічки, в якій всі світлодіоди однаково реагують на сигнал з RGB-контролера, в адресній LED-стрічці кожен світлодіод отримує індивідуальну команду управління. В результаті користувач може максимально точно підбирати відтінок для кожного світлодіода, створювати світлові ефекти і збирати матриці з 16 млн кольорів (рис. 2.9).

Рисунок 2.9 – Адресна світлодіодна стрічка WS2812B

Світлодіодна стрічка зазвичай має ширину 10-12 мм і довжину до 5 метрів або більше. Її можна обрізати до певної довжини, використовуючи лише пару ножиць уздовж ліній розрізу, розташованих через кожні декілька сантиметрів.

Окремі світлодіоди монтується вздовж стрічки, як правило, з щільністю 60-120 світлодіодів на метр. Колір світла та якість окремих світлодіодів визначають загальний колір світла та якість світлодіодної стрічки.

Зворотна сторона світлодіодної стрічки містить попередньо нанесений двосторонній клей. Необхідно просто зняти підкладку і закріпити світлодіодну стрічку практично на будь-якій поверхні. Оскільки друкована плата розроблена як гнучка, світлодіодні стрічки можна монтувати на вигнутих і нерівних поверхнях.

Яскравість світлодіодних стрічок визначається за метричними люменами. На відміну від ламп розжарювання, різні світлодіодні стрічки можуть мати різний рівень ефективності, тому номінальна потужність не завжди має значення для визначення фактичної світлової потужності.

Яскравість світлодіодної стрічки зазвичай описується в люменах на метр.

Якісна світлодіодна стрічка повинна забезпечувати щонайменше 1500 люмен на метр, що забезпечує приблизно таку ж кількість світлового потоку на метр, що й традиційна люмінесцентна лампа T8.

Яскравість світлодіодної стрічки в першу чергу визначається трьома факторами:

- світловіддача та ефективність на світлодіодний випромінювач;
- кількість світлодіодів на метр;
- потужність світлодіодної стрічки на метр.

Світлодіодна стрічка без специфікації яскравості в люменах є червоним прапорцем. Для побудови зображення шляхом обертання було обрано покращену версію світлодіодної стрічки – адресну світлодіодну стрічку.

Адресна світлодіодна стрічка WS2812B - вдає із себе гнучку, струмопровідну плату, на якій розташовані світлодіоди SMD5050 з вбудованими мікросхемами WS2812. На відміну від звичайної RGB-стрічки, в якій всі світлодіоди однаково реагують на сигнал з RGB контролера, в програмованій стрічці, кожен світлодіод управляється індивідуально сигналом з мікросхеми. Програмовані світлодіодні стрічки відрізняються яскравими, насиченими кольорами світіння. Це дозволяє підібрати ідеальний відтінок для кожного світлодіода, створювати і управляти світловими ефектами з 16 мільйонів відтінків кольорів (рис. 2.10).

Рисунок 2.10 – Зображення стрічки WS2812B

Переваги WS2812B.

1. Компактність: піксель містить в своєму корпусі розміром всього 5 x 5 мм світлодіоди і драйвери для них.

2. Простота управління: піксель управляється за допомогою простого послідовного інтерфейсу, який легко реалізується як за допомогою програмних засобів, так і за допомогою апаратних інтерфейсів.

3. Управління всього по одній лінії.
4. Велика кількість включених послідовно пікселів.
5. Відносно невелика вартість.

2.4 Мікро двигун

Двигун – це індуктивне навантаження, яке у момент відключення створює індуктивні викиди. У мотора є щітки, які є джерелом іскор і перешкод за рахунок тієї самої індуктивності котушки. Сам мотор споживає енергію не дуже рівномірно, що може стати причиною перешкод по лінії живлення, а пусковий струм мотора так взагалі сильно більший за робочий струм, що гарантовано просадить слабе живлення при запуску. Усі чотири джерела перешкод можуть призводити до різних глюків у роботі пристрою аж до спрацьовування кнопок на цифрових пінах, наведення перешкод на аналогових пінах, раптового зависання і навіть перезавантаження мікроконтролера або інших залізних пристроїв у зборі. Відсікти індуктивний викид з двигуна можна за допомогою найпростішого діода, чим потужніший мотор, тим потужніший потрібен діод, тобто на більш високу напругу і струм. Діод ставиться зустрічно паралельно мотору, і що ближче до корпусу, то краще. Так само рекомендується чинити з електромагнітними клапанами, соленоїдами, електромагнітами і взагалі будь-якими іншими котушками. Логічно, що діод потрібно ставити тільки в тому випадку, якщо двигун або котушка керується в один бік[10]. Важливі моменти.

1. Під час роботи з драйвером та керуванням у обидві сторони діод ставити не потрібно.
2. При керуванні сигналом ШИМ рекомендується ставити швидкодіючі діоди (наприклад серії 1N49xx) або діоди Шоттки (наприклад серії 1N58xx).
3. Максимальний струм діода повинен бути більшим або дорівнює максимальному струму двигуна.
4. Захисний діод, що приймає зворотний викид ЕРС самоіндукції, також називається шунтуючим діодом, снаббером, flyback діодом.

5. У природі існують мосфети із вбудованим захисним діодом. Цей діод є окремим елементом і такий мосфет має нестандартний корпус.

Щітки мотора, що іскряться, особливо старого і розбитого, є сильним джерелом електромагнітних перешкод, і тут проблема вирішується установкою керамічних конденсаторів з ємністю 0.1-1 мкФ на висновки мотора. Такі ж конденсатори можна поставити між кожним виведенням та металевим корпусом, це ще сильніше погасить перешкоди. Для паяння до корпусу потрібно використовувати потужний паяльник і активний флюс, щоб залудитися і припаятися якнайшвидше, не перегріваючи двигун.

Мотор споживає струм не дуже рівномірно, особливо під час розгону або в умовах змінного навантаження на вал, що проявляється у вигляді просадок напруги живлення всієї схеми. Біди з живленням вирішуються установкою ємних електролітичних конденсаторів по живленню, логічно ставити їх потрібно дуже близько до драйвера. Напруга має бути вище напруги живлення, а ємність вже підбирається за фактом. Почати можна з 470 мкФ і підвищувати, доки не стане добре.

В окремих випадках критичними є навіть наведення від дротів моторів, що живлять, особливо при управлінні ШИМ потужними моторами і управлінні потужними кроковиками в верстатах. Такі наведення можуть створювати сильні перешкоди для чутливих електронних компонентів, що працюють поруч, на аналогові ланцюги, що приводять перешкоди на лінії вимірювання АЦП і звичайно ж на радіозв'язок. Захиститися від таких перешкод можна за допомогою екранування силових проводів: екрановані силові проводи не завжди вдається купити, тому достатньо обмотати звичайні фольгою проводи і підключити екран до GND живлення силової частини.

Польовий транзистор, він же мосфет, дозволяє керувати швидкістю обертання двигуна за допомогою ШИМ сигналу. При використанні мосфету обов'язково потрібно ставити діод, інакше індуктивний викид з двигуна дуже швидко вб'є транзистор. Швидкість двигуна можна задавати за допомогою ардуїновського аналоговогопису (пін, швидкість). Найкраще управляти двигуном за допомогою спеціального драйвера, вони бувають різних форм і

розмірів і розраховані на різну напругу і струм, але управляються практично однаково.

Колекторний електродвигун складається з ротора і статора. Хоча також існують деякі варіації, двигунів зі стаціонарним ротором і обертовим статором. Статор складається з пари постійних магнітів з протилежним розташуванням полюсів, а ротор – з поперечини, обмотаним дротом в протилежних напрямках з кожного боку. Коли обидві котушки підключені до джерела живлення, вони діють як електромагніти з протилежними полярностями (рис. 2.11).

Рисунок 2.11 – Колекторні двигуни постійного струму

Електродвигуни працюють за рахунок сил Лоренца, які виникають при проходженні електричного струму через обмотки, розташовані в магнітному полі. Вплив цих сил змушує ротор обертатися навколо своєї осі. Крутний момент, створюваний силою Лоренца, є векторним, що означає, що коли полюса електромагнітів, утворених обмотками ротора, вирівняні з протилежними полюсами магнітів статора, сила падає до нуля, а ротор припиняє обертання (рис. 2.12).

Рисунок 2.12 – Принцип дії колекторного двигуна

Однак зміна напрямку струму в обмотках призведе до зміни полярності електромагнітів. Сила буде відновлена і ротор відновить рух. Якщо ця зміна буде відбуватися кожного разу, при проходженні вертикалі статора, ротор буде продовжувати обертатися і виконувати корисну роботу.

Для зміни напрямку струму з контрольованою частотою, колекторних двигунів постійного струму вимагають колектор. Колектор – це розділене на

сегменти кільце відповідним чином підключений до кожної з обмоток ротора. Коли ротор обертається – теж відбувається і з колектором. Для того щоб підвести струм до колектора до нього з протилежних сторін притискається пара нерухомих щіток. Коли колектор повертається, кожен сегмент колектора послідовно контактує спочатку з одною щіткою / джерелом струму, а потім з іншою. В результаті струм в роторних котушках змінюється кожен раз при повороті ротора на 180° , підтримуючи обертання двигуна.

Альтернативою є безколекторні двигуни постійного струму або двигуни з електронним комутатором. Двигуни є синхронні двигуни з постійними магнітами. Вони можуть працювати як серводвигуни, а також як крокові двигуни. Це визначення також включає двигуни з перемикальним опором. Постійні магніти встановлені на роторі, а статор складається з ламінованої рами з котушками. В результаті ротор не потребує будь-якої проводки, і двигун не потребує колектора і щітки (рис. 2.13).

Рисунок 2.13 – Безколекторні двигуни постійного струму

Оскільки в безколекторних двигунах не застосовуються щітки і колектори, вони більш компактні, ніж колекторні двигуни. Вони забезпечують більш високу продуктивність в одному типорозмірі. Відсутність щіток знижує необхідність обслуговування і дозволяє ротору обертатися на більш високих швидкостях. Відсутність тертя вирівнює криву швидкості, усуває ймовірність іскріння і знижує електромагнітні перешкоди. Переміщення теплогенеруючих обмоток назовні спрощує тепловідвід. Цей підхід також знижує інерційність ротора, дозволяючи сервомоторам забезпечувати кращий динамічний відгук. Відсутність падіння напруги на щітках також підвищує ефективність безколекторних двигунів[11].

З іншого боку, безколекторні двигуни складніше, ніж їх колекторні аналоги. Використання вбудованої електроніки значно збільшує їх вартість.

Щоб повною мірою зрозуміти чим відрізняється колекторний двигун від

безколекторного, варто зважити всі переваги і недоліки обох типів. Щіткові електродвигуни постійного струму є кращим рішенням в області управління рухом. Вони економічні і прості у використанні. Оскільки їм не потрібно вбудована електроніка, вони можуть витримувати екстремальні умови. За умови, що щітки обрані правильно і своєчасно обслуговуються, щіткові двигуни постійного струму можуть служити тривалий час. Вони добре підходять для застосування в пристроях з помірними і низькими швидкостями.

Вибір типу двигуна обумовлюється вимогами, які до нього виставляються. Проект з обмеженим бюджетом і з помірними вимогами до характеристик двигуна може відмінно бути реалізований з використанням колекторного двигуна постійного струму. Якщо для проекту більш важливими є продуктивність і робочий цикл безколекторний двигун може бути кращим рішенням. Оригінальний виробник обладнання і кінцеві користувачі повинні враховувати не тільки можливості двигуна, але і можливості свого персоналу по інсталяції та обслуговування обладнання. Ефективне технічне рішення може бути прийнято тільки при обґрунтованому виборі обладнання.

2.5 Модуль датчика Холла

Датчик Холла - модуль KY-003 для Arduino виконаний на базі елемента "44E" і по суті є магнітним перемикачем. Якщо поруч з ним немає магнітного поля, то на сигнальному виході датчика є високий рівень і навпаки, якщо поле присутній, то на виході низький рівень напруги, таким чином сенсор має цифровий TTL вихід. Існують також аналоговий «49E» і комбінований датчики Холла. На платі є світлодіод, який горить коли є магнітне вплив (рис. 2.14).

Рисунок 2.14 – Модуль датчика Холла

Датчик ефекту Холла – вимірювальний перетворювач для вимірювання величини магнітного поля. Принцип роботи датчика заснований на ефект Холла і його вихідна напруга прямо пропорційно напруженості магнітного поля. Це явище було відкрито американським фізиком Едвіном Холлом у 1879 році.

Датчики ефекту Холла використовуються для безконтактного визначення, позиціонування, виявлення швидкості та визначення струму. Часто датчик Холла поєднується з виявленням порога, і діє як перемикач і називається перемикачем Холла.

Датчики Холла зазвичай використовуються для вимірювання швидкості обертання коліс та валів, наприклад, для синхронізації запалення двигуна внутрішнього згорання, тахометрів та антиблокувальної гальмівної систем. Вони використовують у вентильних електродвигунах постійного струму виявлення положення постійного магніту.

Пристрої на ефекті Холла несприйнятливі до пилу, бруду та води. Ці характеристики роблять пристрої на ефекті Холла краще для визначення положення, ніж альтернативні засоби, такі як оптичні та електромеханічні вимірювання.

Коли електрони проходять крізь провідник, створюється магнітне поле. Таким чином можна створити безконтактний датчик струму. Пристрій має три термінали. Напруга датчика прикладається до двох клем, а третій забезпечує напругу, пропорційну вимірюваному струму. Це дає кілька переваг: ніякого додаткового опору - шунту, необхідного для найбільш поширеного методу вимірювання струму, не потрібно первинного ланцюга. Крім того, напруга, яка присутня в лінії, яка повинна бути виміряна, не передається на датчик, що підвищує безпеку вимірювального обладнання.

2.6 Bluetooth модуль

Bluetooth — один із найпоширеніших протоколів бездротового зв'язку на малій відстані для персональних електронних пристроїв. З його допомогою

працюють різні пристрої введення та управління, аудіопристрою.

На тлі інших протоколів Bluetooth виділяється насамперед завадостійкістю та простотою. Від не менш поширеного Wi-Fi Bluetooth відрізняється насамперед низьким енергоспоживанням, що робить його доступним для автономних пристроїв. Завдяки цьому набув повсюдного поширення.

Найчастіше для саморобок використовуються BT-модулі HC-05 та HC-06. Їх легко знайти у продажу. HC-05 відрізняється від HC-06 тим, що йому доступні обидва режими роботи: ведений (master) і ведучий (slave). А ось HC-06 працює тільки веденим, тобто він не здатний знаходити інші пристрої та самостійно встановлювати з ними зв'язок, але це можна виправити.

Нерідко у проектах виникає потреба дистанційного керування або передачі даних з телефону чи іншого пристрою. Одним із найпопулярніших та зручних способів є обмін даних через Bluetooth. Для зв'язку плати Arduino та комп'ютера використовується інтерфейс UART (Serial). Оскільки будь-яка плата Arduino має хоча б один послідовний порт UART, для підключення Bluetooth модуля не потрібні спеціалізовані бібліотеки та схеми.

Найпопулярнішими модулями є пристрої на основі чіпа BC417. Ця серія називається HC. Модулі HC-03 і HC-05 можуть бути і сервером з'єднання, і клієнтом, вони мають широкий набір AT команд.

Для підключення до ПК потрібні модуль Bluetooth, плата Arduino, з'єднувальні дроти та комп'ютер. Скетч для керування платою Arduino через смартфон і комп'ютер буде однаковим, тому що в обох випадках дані в мікроконтролер надходять за протоколом UART.

Обидва модулі є дві спаяні плати. Одна з них – заводська з мікросхемою, інша потрібна для саморобних пристроїв, вона оснащена ніжками GPIO зі стандартним кроком 2,54 мм та стабілізатором напруги. Модуль HC-05 коштує трохи дорожче, але має більшу кількість корисних робочих функцій (рис. 2.15).

Рисунок 2.15 – Bluetooth модулі HC-05 і HC-06

Швидкість передачі AT команд за замовчуванням для HC-05 дорівнює 38400, для HC-06 - 9600. Важливим моментом є те, що в кінці команд AT для HC-05 повинні бути символи CRLF.

Всі існуючі типи модулів Bluetooth мають свої особливості, але за функціями та дією вони схожі. Одним із видів модулів є Bluetooth HC 06. З боку Arduino модуль виглядає як звичайний послідовний інтерфейс, тому можна відразу налагодити взаємодію з пристроєм на комп'ютері.

Основні характеристики модуля:

- живлення 3,3 – 6 В;
- максимальна вхідна напруга 5;
- максимальний струм 45 мА;
- швидкість передачі даних 1200-1382400 бод;
- робочі частоти 2,40 ГГц – 2,48 ГГц;
- підтримка специфікації bluetooth версії 2.1;
- мале споживання енергії;
- високий рівень захисту даних;
- дальність зв'язку 30 м;
- для підключення до смартфона використовуються такі дані: пароль «1234», швидкість передачі даних 9600, ім'я модуля HC-06.

Модуль має такі контакти:

- VCC, GND – плюс та мінус харчування;
- RX та TX – приймач та передавач;
- MCU-INT – виводить статус;
- Clear (Reset) – скидання та перезавантаження модуля.

Модуль HC-06 використовується тільки в режимі Slave, тобто він не може самостійно підключатися до інших пристроїв Bluetooth. Усі налаштування для підключення «Пароль, швидкість передачі даних» можна змінити за допомогою AT-команд.

3 ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ ПРОЕКТУ

3.1 Середовище програмування Arduino IDE

Програмну частину проекту було виконано в середовищі розробки від компанії виробника плати – Arduino IDE. Arduino IDE - крос-платформений додаток, розроблений на C і C ++. Він використовується для написання і завантаження програм на Arduino-сумісні плати, а також на плати інших виробників. Середовище розробки Arduino складається з вбудованого текстового редактора програмного коду, області повідомлень, вікна виведення тексту, панелі інструментів з кнопками, часто використовуваних команд і декількох меню. Для завантаження програм і зв'язку середовище розробки підключається до апаратної частини Arduino (рис. 3.1).

Рисунок 3.1 – Інтерфейс середовища програмування Arduino IDE

Для завантаження скетчу на плату Arduino Nano бажано відключити плату від модулів та живлення та виконати декілька послідовних дій.

1. Підключити Arduino Nano до комп'ютера через програматор USBAsp.
2. Запустити в Arduino IDE потрібний скетч.
3. У вкладці «Інструменти» обрати плату, установлений на ній процесор та COM-порт підключення, відмінний від COM3.
4. У вкладці «Скетч» вибрати «Загрузка через програматор» або скористатися комбінацією клавіш «Ctrl+Shift+U».
5. Після успішної загрузки скетчу середовище розробки сповістить про це написом в нижній частині відкритого вікна.

Після виконання алгоритму загрузки скетчу можна підключити плату до

установки та живлення.

У жовтні 2019 року організація Arduino почала надавати ранній доступ до нової Arduino Pro IDE з налагодженням та іншими розширеними функціями.

Середовище розробки розповсюджується вільно. Інсталювачі Arduino IDE можна завантажити з сайту для операційних систем Windows, Mac OS, Linux. Як компілятор середовище використовує відомий `avr-gcc`, а значить, що «з коробки» користувач отримує підтримку оригінальних плат з архітектурою AVR та аналогів.

Інтерфейс у IDE для Arduino гранично лаконічний: меню та функціональні кнопки – зверху, редактор – у центрі, лог роботи компілятора та завантажувача – внизу. У робочій області відкриваються вкладки, але вони в даному випадку мають особливе значення. По-перше, не є окремими проектами: вони розбивають код інструкції, що знаходиться в одній папці, на смислові частини. По-друге, читаються і, головне, компілюється зліва направо. І, по-третє, розташовуються в алфавітному порядку[15].

3.2 Програмна реалізація системи управління

Першим кроком в побудові алгоритму маніпулювання адресною світлодіодною стрічкою є ініціалізація констант. Для цього буде використовуватися зручна директива `#define`.

`#define` - це зручна директива, яка дозволяє дати ім'я константі перед тим як програма буде скомпільована. Певні цією директивою константи не займають програмної пам'яті, оскільки компілятор замінює всі звернення до них їх значеннями на етапі компіляції, відповідно вони служать виключно для зручності програміста і поліпшення читаності тексту програми[13].

Варто згадати про деякі небажані ефекти, які можуть мати місце при використанні директиви `#define`. Наприклад, якщо ім'я константи, заданої за допомогою директиви `#define` включити в ім'я іншої константи або змінної, то воно буде замінено на своє значення (лістинг 3.1).

Лістинг 3.1 – Ініціалізація констант та пінів підключення периферії

```
#define NUM_LEDS 72
#define BRIGHTNESS 200
#define MOTOR_MAX 180
#define RES 10
#define COEF 0.3
#define PODGON 1.3
#define OFFSET 240
#define NUM_FRAMES 8
#define FRAME_RATE 60

#define PIN 4
#define MOS 3
#define BTN1 8
#define BTN1_G 6
```

В даній частині коду здійснюється присвоювання незмінним константам їх єдиного значення. На етапі компіляції кожне звернення до них буде замінено на їх значення. Наприклад використана світлодіодна стрічка містить 72 світлодіода, це значення залишиться незмінним. Максимальна яскравість (BRIGHTNESS) виставлена в найбільшу величину - 200. Швидкість двигуна очікується постійною, вона буде використовуватись для розрахунку поворотів малюнка або анімації та складає 180 обертів на хвилину. Вбудована флеш пам'ять платформи Arduino Nano на чіпі ATmega328 складає 32 кБ., отже кожна анімація може бути розділена на 8 послідовних кадрів (NUM_FRAMES). Дослідивши технічну характеристику адресної світлодіодної стрічки WS2812B, враховуючи частоту перерисовки одного світлодіода, було вираховано, що між зміною кадра анімації світлодіодна стрічка може перерисуватись 60 разів (FRAME_RATE).

В якості констант також були ініціалізовані піни підключення периферійних засобів. Піном світлодіодної ленти був обраний 4 цифровий пін. Кнопка вмикання/вимикання встановлена на 6 та 8 піни.

Після ініціалізації незмінних констант в даній кваліфікаційній роботі було проведено підключення необхідних файлів та бібліотек в проект (лістинг 3.2).

Лістинг 3.2 – Підключення використаних бібліотек та файлів

```

//#include "homer.h"
#include "sonic.h"
//#include "sponge.h"

#include <TimerOne.h>
#include "Adafruit_NeoPixel.h"

#include "gamma.h"

```

Директива `#include` використовується для включення сторонніх бібліотек та файлів в ваш скетч. Це дає доступ до великої кількості стандартних бібліотек C, і бібліотек написаних спеціально для Arduino. Бібліотекою називають групи попередньо написаних функцій.

Зверніть увагу, що `#include`, так само як і `#define`, не вимагає точки-коми в кінці, якщо ж її додати, компілятор видасть критичну помилку. Для демонстрації роботи проекту було підготовлено декілька анімацій з відомих мультфільмів. Для роботи з адресною світлодіодною стрічкою використовується бібліотека «Adafruit_NeoPixel» та «gamma», а для коректної роботи з часом – бібліотека «TimerOne». Завдяки директиві `#include` їх було успішно підключено.

Для контролювання адресної світлодіодної стрічки за допомогою бібліотеки «Adafruit_NeoPixel» необхідно створити об'єкт класу «Adafruit_NeoPixel», задати його ім'я та передати в параметри кількість пікселів стрічки та пін підключення. Після цього слідує ініціалізація додаткових змінних, що полегшують роботу алгоритма.

Як зазначалося, у блоці коду функції `setup()` всі дії та команди виконуються один раз після запуску програми. Виконується настройка режимів роботи портів, підключення їх до модулів та запуску використовуваних бібліотек (лістинг 3.3).

Лістинг 3.3 – Структура функції `setup()`

```

void setup() {
  Serial.begin(115200);
  strip.begin();
  strip.setBrightness(BRIGHTNESS);

  pinMode(BTN1, INPUT_PULLUP);
  pinMode(BTN2, INPUT_PULLUP);
  pinMode(MOS, OUTPUT);

  digitalWrite(BTN1_G, LOW);

```

```

attachInterrupt(0, hall, RISING);
Timer1.initialize();
Timer1.stop();

strip.clear();
strip.show();
}

```

Для початку роботи з послідовним портом вводу та виводу інформації використовується метод `begin` класу `Serial`. В якості параметрів функції передається швидкість роботи з портом в бодах. Бод - одиниця виміру символної швидкості, кількість змін інформаційного параметра несучого періодичного сигналу в секунду. Названа по імені Еміля Бодо, винахідника коду Бодо - кодування символів для телетайпів. В параметр функції передається найбільше значення в 115200 бод.

У об'єкта `strip`, відповідального за роботу з адресною світлодіодною стрічкою через оператор «.» визваний метод `begin`, що означає запуск підключеної бібліотеки. Також була задана яскравість стрічки завдяки функції `setBrightness()`.

Функція `pinMode()` дозволяє обрати для будь-якого піна режим роботи. Для пінів підключення кнопки це значення буде відповідне режиму `INPUT_PULLUP`, оскільки з кнопки ми будемо отримувати натискання. А для контролю розгону двигуна використовується режим роботи піну `OUTPUT`.

Функція `digitalWrite()` подає на цифровий пін значення `LOW(0)` або `HIGH(1)`. Для піну землі кнопки було встановлене початкове значення 0, натискання кнопки буде змінювати `LOW` на `HIGH` та система розпочне виконувати роботу.

Для роботи з перериваннями використовується функція `attachInterrupt()`. Вона задає функцію, яку необхідно викликати при виникненні зовнішнього переривання. Замінює попередню функцію, якщо така була раніше асоційована з перериванням. `attachInterrupt()` приймає в параметри номер переривання, функцію, яку треба виконати коли з'явилося переривання та режим – умову при якій з'явиться переривання[14]. Існують декілька умов появи переривання:

- `LOW` - переривання буде спрацьовувати щоразу, коли на виводі

присутній низький рівень сигналу;

- CHANGE - переривання буде спрацьовувати щоразу, коли змінюється стан виведення;

- RISING - переривання спрацює, коли стан виведення зміниться з низького рівня на високий;

- FALLING - переривання спрацює, коли стан виведення зміниться з високого рівня на низький.

Після цього виконуються ініціалізація та запуск бібліотеки TimerOne та очистка стрічки.

Однією з найважливіших функцій в проекті являється функція setPix(). В цій функції проводиться послідовність дій, що призводить до призначення кожному пікселю індивідуального кольору. Обрані для прикладу анімації знаходяться у вбудованій флеш-пам'яті мікроконтролера, отже в функції setPix() з вбудованої пам'яті витягується колір пікселя глибиною 16 біт, переводиться в 24 бітний формат завдяки бібліотеці Adafruit_NeoPixel тазначається певному пікселю (лістинг 3.4).

Лістинг 3.4 – Алгоритм призначення кольору функцією setPix()

```
void setPix(byte i, int counter, int this_pix) {
  switch (this_frame) {
    case 0: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame1[counter][this_pix]))));
      break;
    case 1: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame1[counter][this_pix]))));
      break;
    case 2: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame2[counter][this_pix]))));
      break;
    case 3: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame3[counter][this_pix]))));
      break;
    case 4: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame4[counter][this_pix]))));
      break;
    case 5: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame5[counter][this_pix]))));
      break;
    case 6: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame6[counter][this_pix]))));
      break;
    case 7: strip.setPixelColor(i,
expandColor(pgm_read_word(&(frame7[counter][this_pix]))));
      break;
  }
}
```

```

    }
}

```

Кожна анімація, що зберігається в пам'яті мікроконтролера розбита на 8 кадрів(фреймів). Послідовне вмикання фреймів анімації утворює ефект анімації. В функції setPix() відстежується зміна фреймів, завдяки оператору switch-case обирається потрібний фрейм і за допомогою strip.setPixelColor() певний світлодіод отримує значення свого кольору.

Якщо функція setPix() створювалася для призначення пікселю індивідуального кольору, то функція animation() вирішує, якому саме пікселю та який давати колір. Алгоритм реалізовано завдяки використанню декількох лічильників та циклів «for». Також в функції animation() напряду використовується раніше написана функція setPix() (лістинг 3.5).

Лістинг 3.5 – Алгоритм відтворення анімації

```

void animation() {
    timer_isr = false;
    strip.clear();

    byte this_pix = 0;
    for (int i = NUM_LEDS / 2 - 1; i >= 0; i--) {
        setPix(i, counter1, this_pix);
        this_pix++;
    }
    this_pix = 0;
    for (int i = NUM_LEDS / 2 ; i < NUM_LEDS; i++) {
        setPix(i, counter2, this_pix);
        this_pix++;
    }

    counter1++;
    counter2++;
    if (counter2 > n_segm) counter2 = 0;
    if (counter1 > n_segm) counter1 = 0;

    strip.show();
}

```

На початку функції виконується очистка стрічки від ввімкнутих пікселів. Як лічильники виступають змінні: counter1, counter2, this_pix. counter1 та counter2 виступають в якості лічильників пікселів першої та другої половини стрічки. This_pix відповідає за позицію обраного пікселя. Для обробки пікселів з обох половин стрічки було написано два цикла for, в яких визивається раніше побудована setPix().

Останньою функцією в даному проекті є обов'язкова для середовища програмування функція loop(). В основному в цьому блоці кода відбуваються основні обчислення, визови функцій та обробка подій. Блок коду написаний в функції loop() буде виконуватись нескінченно в циклі, доки його не зупинити, або не вимкнути живлення плати (лістинг 3.6).

Лістинг 3.6 – Структура обов'язкової функції loop()

```

void loop() {
  if (hall_isr) {
    hall_isr = false;
    counter1 = 0 + offset;
    counter2 = n_segm / 2 + offset;
    if (counter2 > n_segm) counter2 = counter2 - n_segm;

    count_frame++;
    if (count_frame >= FRAME_RATE) {
      count_frame = 0;
      this_frame = ++this_frame % NUM_FRAMES;
    }

    animation();
    period_f = (float)new_period * COEF + (float)period_f * ((float)1 -
COEF);
    Timer1.setPeriod(period_f / n_segm * PODGON);
    Timer1.restart();
  }

  if (timer_isr) {
    timer_isr = false;
    animation();
  }

  if (!digitalRead(BTN1)) {
    motor_state = !motor_state;
    if (motor_state) {
      for (int i = 0; i < MOTOR_MAX; i++) {
        analogWrite(MOS, i);
        delay(20);
      }
      Timer1.attachInterrupt(timer_interrupt);
    } else {
      for (int i = MOTOR_MAX; i > 0; i--) {
        analogWrite(MOS, i);
        delay(20);
      }
      analogWrite(MOS, 0);
      Timer1.stop();
      Timer1.detachInterrupt();
      delay(500);
      timer_isr = false;
      strip.clear();
      strip.show();
    }
  }
}
}

```

Функція розпочинається з вибору фрейму та швидкості обертання. Завдяки оператору `if` відбувається початкова ініціалізація лічильників та змінних з урахуванням коефіцієнтів відхилення. Далі визивається функція `animation()` та розраховується період обертання установки по виведеній формулі. Значення таймера встановлюється з урахуванням розрахованого періоду обертання та відбувається його перезапуск.

Функція закінчується реалізацією обробника натискань кнопки. При вимиканні системи відбувається повільне зниження швидкості до повного відключення двигуна, вимикання стрічки та таймерів.

3.3 Додаток Pixel

Приклади анімацій повинні зберігатися в пам'яті мікроконтролера, тож для кожної з них було створено файл з розширенням `.h` в якому повинна зберігатися картинка. На просторах інтернету існує додаток для розбиття картинки формату `bmp` у масив кольорів полярних координат. Саме його і буде записано у пам'ять мікроконтролера (рис. 3.2).

Рисунок 3.2 –Інтерфейс додатку «Pixel»

Додаток може розбити будь яку картинку в форматі `bmp` на кольори з обраною вами глибиною. В інтерфейсі програми також можна виставити кількість сегментів, розмір сегмента, кількість градусів в сегменті, режими малювання. Кнопка «Открыть» дозволяє обрати картинку з існуючих в дозволеному форматі (рис. 3.3).

Рисунок 3.3 – Демонстрація роботи з додатком

Після обрання параметрів розбиття і натискання кнопки «Рисовать» відбувається розбиття картинки на кольорові координати. Їх можна побачити в нижній частині додатку у вікні виводу інформації. Отриманий масив необхідно скопіювати та вставити у файл проекту з розширенням .h. (лістинг 3.7).

Лістинг 3.7 – Масив кольорів полярних координат в файлі «.h»

```
const uint16_t frame0[][36] PROGMEM={
  {0xE4EF, 0xE510, 0xED10, 0x9AC7, 0x4928, 0x5ADA, 0x631D, 0x6B5D, 0x52FD, 0x39F4, 0x9493,
  0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
  0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000 }, //10 + 0
  .....
  .....
  {0xE4EF, 0xE510, 0xED10, 0x9AC7, 0x4928, 0x5ADA, 0x631D, 0x6B5D, 0x52FD, 0x39F4, 0x9493,
  0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000,
  0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000, 0x0000 }, //370 + 0
};
```

3.4 Апаратні рішення та підключення установки

В данній кваліфікаційній роботі велику роль грає вага проектованого приладу, тож за будь-якою можливістю необхідно підбирати матеріали з низькою вагою. Оскільки використовується мікро двигун для робоплатформ на 12В та потужністю 1 кілограм на сантиметр, тож необхідно максимально полегшити вагу стрічки, що обертається.

Світлодіодна стрічка буде закріплена на пластиковому профілі з довжиною 500 мм. Обраний матеріал та його форма дає світлодіодній стрічці гнучкість та простоту в експлуатації. Від металевої фіксації стрічки було вирішено відмовитися, оскільки існував шанс появи перешкод у логічному сигналі стрічки. Металеве кріплення очікує від двигуна більшої потужності. Стрічка має 3 контакти – VCC (умовний плюс живлення 5В), GND (умовний

мінус живлення), та S (умовний логічний сигнал стрічки). Дроти 5В та GND підключені до стрічки з одного боку, логічний сигнал було підключено з іншого (рис. 3.4).

Рисунок 3.4 – З'єднання світлодіодної стрічки та дротів живлення

Важливим рішенням було об'єднання конструкції, що обертається, та передача сигналів управління та живлення стрічки через музикальний стерео-аудіо штекер 6.3 мм. Такий штекер має 3 контакти, та металевий корпус, що забезпечує його міцність. До гнізда даного штекера на контакти VCC та GND було вмонтовано конденсатор для більш плавної роботи стрічки.

Обертати побудовану конструкцію зможе звичайний китайський мотор з редуктором. Такі мотори дуже популярні в робототехніці та моделюванні і живляться від 12В. Оскільки вся система буде живитися від блоку живлення старого телефона на 5В, отже для двигуна необхідно придбати підвищуючий перетворювач напруги та за допомогою мультиметра встановити значення на 12В (рис. 3.5).

Рисунок 3.5 – Безколекторний мікро-двигун для робоплатформ

Для обрання правильної швидкості оновлювання зображення на стрічці необхідно знати кількість оборотів двигуна в хвилину. Для відстеження оберту на великій швидкості було обрано цифровий модуль датчика Холла.

Модуль датчика Холла – це плата з розпаяним на ній уніполярним датчиком Холла 3144 серії. Сам датчик А3144 вже містить: стабілізатор напруги, детектор магнітного поля, компаратор напруги і ключ з відкритим колектором (рис. 3.6).

Рисунок 3.6 – Кріплення модуля датчика Холла

Всі деталі проекту були оснащені перемичками мама або тато для більш простого монтажу та швидкої заміни деталей. Обраний спосіб з'єднання також дозволив уникнути замикань в ланцюзі. Плату Arduino було вручну перепаяно та оснащено монтажними ніжками для забезпечення з'єднання з іншими деталями проекту через перемички (рис. 3.7).

Рисунок 3.7 – Розпаяна плата Arduino та універсальні перемички

Всі компоненти системи окрім двигуна будуть живитися від постійного струму в 5В, що видає стандартний блок живлення для сучасних смартфонів. Двигун потребує більшої напруги, тож він буде живитися від джерела постійного струму на 12В. Для блоку живлення на 5В було спеціально спаяно перехідник з перемичок «МАМА», оскільки від нього будуть живитися стрічка, плата та датчик Холла. Таким чином зменшується шанс на короткі замикання контактів «+» та «-» (рис. 3.8).

Плата Arduino підключається до живлення по пінам «5В» та «GND». До Arduino через перемички будуть підключені 2 дроти живлення, дріт з модуля датчика Холла та дріт з адресної світлодіодної стрічки. Запуск установки буде відбуватися по кнопці, підключеної до плати.

Так як світлодіодна стрічка буде обертатись, тож з'єднати її напряму з джерелом живлення та платою буде неможливо. Для вирішення проблеми буде

використаний набір з кутового стерео-аудіо штекера та підходящого до нього гнізда. Штекер має 3 контакти, котрі будуть використані для підключення живлення та логічного дроту. Також саме до нього буде закріплений пластиковий профіль зі стрічкою.

Передавання сигналу по поверхні, що обертається стає можливим при наявності підходящого приймача, котрим буде виступати стерео-аудіо гніздо. До нього напряму буде підключене живлення та дріт з плати Arduino. Між контактами живлення закріплений конденсатор, оскільки адресна світлодіодна стрічка чутлива до перепадів напруги. Такі перепади можуть викликати вигорання запобіжного світлодіода.

Модуль датчика Холла буде живитися від джерела постійного струму на 5В, та підключений до плати через перемичку «Dual female».

Рисунок 3.8 – Ланцюг з'єднання компонентів проекту

Розглянувши особливості реалізації проекту, можна прийти до висновку, що отримана система повинна складатися з максимально простих елементів, що можуть легко змінюватись на аналоги для подальшої модернізації або ремонту.

Запропоновано схему реалізації проекту, що визначається компоновкою обраних елементів системи з урахуванням їх характеристик та правилами користування

ВИСНОВКИ

Виконання кваліфікаційної роботи дозволило дослідити використання плат Arduino в проектах різного ступеня складності. Було спроектовано домашній аналог сучасних голографічних вентиляторів, популярних у рекламодавців. Дослідження роботи типового голографічного вентилятору показало, що на основі плат та середовища програмування Arduino можливо спроектувати бюджетний аналог.

Як серце установки виступає плата Arduino моделі Nano з мікроконтролером від фірми Atmel моделі ATmega328P. Обрання Arduino як контролера дозволило виявити ряд переваг, таких як: доступність модулів та датчиків, простота програмування та з'єднання, велика кількість документації розробки під Arduino.

За допомогою програмної складової та алгоритму Arduino з легкістю маніпулює кожним світлодіодом на стрічці. Робота всіх компонентів як єдиного пристрою дозволяє спостерігати за плавністю роботи та своєрідною магією світла, що утворює обране зображення.

Для подальшої модернізації проекту можна провести заміну деяких елементів системи. Найбільше потребує заміни мікро-двигун, він не видає очікуваної потужності. Також необхідно модернізувати механізм обертання в вигляді гнізда та штекера. Оскільки штекер складається з трьох незалежних частин, що тільки стиснуті разом пластиковими перемичками, то при високих оборотах циліндри штекера починають прокручуватись, що впливає на якість картини. Реалізація виводу зображення на дисплей завдяки бездротовій мережі або Bluetooth дозволить додати більше функціональності та зручності при користуванні.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Изучаем Arduino. Инструменты и методы технического волшебства [Электронный ресурс] / Режим доступа: <https://arduino.ua/docs/AOU164.pdf> - Загл. 3 экрану.
2. Инструкция по эксплуатации AVR [Электронный ресурс] / Режим доступа: <http://ww1.microchip.com/downloads/en/DeviceDoc/AVR-Instruction-Set-Manual-DS40002198A.pdf> - Загл. 3 экрану.
3. Ревич Ю.В. Занимательная электроника [Текст] / Ревич - «БХВ-Петербург», 2015.
4. Трамперт В. AVR-RISC Микроконтроллеры [Текст] / Трамперт - «ДМК Пресс», 2006.
5. Баранов В.Н. Применение Микроконтроллеров AVR: схемы алгоритмы программы [Текст] / В.Н. Баранов - «Додэка-XXI», 2006.
6. Соммер Улли. Программирование микроконтроллерных плат Arduino/Freduino [Текст] / Улли Соммер - «БХВ-Петербург», 2012.
7. Francis Perea. Arduino Essentials [Текст] / Francis Perea - «Packt Publishing», 2015.
8. Монк Саймон. Практическая электроника [Текст] / Саймон Монк - «Диалектика-Вильямс», 2012.
9. Офіційний сайт компанії Arduino [Электронный ресурс] / Режим доступа: <https://www.arduino.cc/> - Загл. 3 экрану.
10. Бесколлекторные двигатели [Электронный ресурс] / Режим доступа: https://rep.bntu.by/bitstream/handle/data/41623/Beskollektornye_dvigateli.pdf?sequence=1&isAllowed=y – Загл. 3 экрану.
11. Ревич Ю.В. Азбука электроники. Изучаем Arduino [Текст] / Ревич - «Издательство АСТ», 2009.
12. 3D голографічний дисплей [Электронный ресурс] / Режим доступа: <https://rozetka.com.ua/216192937/p216192937/?gclid=CjwKCAjw26H3BRB2EiwAy32zhc-NRyX5Oqk3vJ1on9qrTPuX69fFE5YyZCkYF> – Загл. 3 экрану.

13. Адресная светодиодная лента [Электронный ресурс] / Режим доступа: https://alexgyver.ru/ws2812_guide/ – Загл. 3 экрану.

14. Цифровой модуль датчика Холла "KY-003" [Электронный ресурс] / Режим доступа: <https://arduino.ua/prod1210-modyl-datchika-holla-a3144> – Загл. 3 экрану.

15. Stereo аудио кутовый штекер [Электронный ресурс] / Режим доступа: <https://perehodniki.ua/p653055907-razem-shteker-stereo.html> – Загл. 3 экрану.