

Міністерство освіти і науки України
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра комп'ютерної інженерії

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри,
канд. техн. наук, доцент

_____ Я. М. Крайник

« __ » _____ 2022 р.

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Програмно-апаратний комплекс для інтерактивної зовнішньої реклами

Спеціальність 123 Комп'ютерна інженерія

123 – КР.1 – 405.21810510

Студент:

_____ А. Г. Качанов

« __ » _____ 2022 р.

Керівник: ст.викладач

_____ І. С. Бурлаченко

« __ » _____ 2022 р.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	4
ВСТУП	5
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ТА СКЛАДОВИХ СИСТЕМИ	7
1.1 Реклама.....	7
1.2 Аналіз основних характеристик конкурентів	8
1.3 Порівняння моделей-конкурентів	15
1.4 Вибір комплектуючих (апаратна база)	16
1.4.1 Мікроконтролер	16
1.4.2 Світлодіодна матриця.....	18
Висновки до розділу 1	22
РОЗДІЛ 2 РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ.....	23
2.1 Детальний огляд світлодіодної матриці	23
2.1.1 Як працює світлодіодна матриця	23
2.1.2 Використання мікросхе МАХ7219 для керування світлодіодною матрицею 8×8.	24
2.1.3 МАХ7219 Конфігурація контактів.....	24
2.1.4 Підключення мікросхе МАХ7219 до світлодіодної матриці .	25
2.1.5 Підключення світлодіодної матриці МАХ7219 до Arduino	26
2.1.6 Про потужність	28
2.1.7 Керування світлодіодною матрицею 8×8 за допомогою МАХ7219 та Arduino	29
2.1.8 Інтерфейс SPI та Специфікація МАХ7219.....	30
2.2 З'єднання всієї світлодіодної матриці разом	31
2.3 Детальний огляд Arduino Uno.....	33
2.4 Порівняння Arduino та Raspberry Pi.....	38
Висновки до розділу 2	43

РОЗДІЛ 3 ПРОГРАМНА ЧАСТИНА КОМПЛЕКСУ	44
3.1 Опис технології та мови програмування	44
3.1.1 Вимоги до синтаксису	46
3.1.2 Опис коду.....	47
3.2 Вибір компонентів АПЗ	50
3.2.1 Бібліотека MD_Parola.h	51
3.2.1 Бібліотека MAX72xx	52
3.2.2 Бібліотека SPI	52
3.2.3 Бібліотека MD_UISwitch.h.....	52
3.3 Опис інтерфейсів АПЗ.....	53
3.3.1 Схеми підключення датчиків до Arduino	53
3.4 Тестування приладу	54
3.4.1 Експериментальне підтвердження	55
3.4.2 Робота приладу.....	55
Висновки до розділу 3	57
ВИСНОВКИ.....	59
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	61
ДОДАТОК А ЛІСТІНГ ПРОГРАМИ	63

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ASCII	–	American Standard Code for Information Interchange
BPI	–	Bytes Per Inch
CAD	–	Computer Aided Design
CAE	–	Computer Aided Engineering
DDL	–	Data Definition Language
DEC	–	Digital Equipment Corporation
EBCDIC	–	Extended Binary Coded Decimal Interchange Code
EPROM	–	Erasable Programmable Read-Only Memory
GSM	–	Global System for Mobile communication
MIPS	–	Millions of Instructions Per Second
RAM	–	Random Access Memory
SQL	–	Structured English QUery Language
UV	–	відповідність між координата на поверхні тривимірного об'єкту (X, Y, Z) і координата на текстурі (U, V).
VFX	–	Visual effects
Гб	–	Гігабайт
ОС	–	Операційна система
ПЗ	–	Програмне забезпечення

ВСТУП

В наш час реклама стала буденним явищем . У сучасному світі реклама використовує всі можливі засоби масової інформації, щоб донести своє повідомлення. Це робиться за допомогою телебачення, друку (газети, журнали, журнали тощо), радіо, преси, Інтернету, прямих продажів, накопичень, розсилок, конкурсів, спонсорства, плакатів, одягу, подій, кольорів, звуків, зображень і навіть людей (підтримка) . Рекламна індустрія складається з компаній, які займаються рекламою, агентств, які створюють рекламу, засобів масової інформації, які передають рекламу, і безлічі людей, як-от редактори копій, візуалізатори, бренд-менеджери, дослідники, креативні керівники та дизайнери, які проходять останню ланку до замовника або одержувач.

Тому зовнішня реклама стала однією з основних методів маркетингу і розробка програмно-апаратного комплексу може принести певну конкуренцію в цю сферу.

Мета: є створення комплексу зовнішньої реклами для того, щоб принести певні новинки та вдосконалити старі методи використання реклами.

Об'єкт: методи визначення якості рекламних компаній; засоби реклами, мікроконтролерні модулі їх оброблення та індикації .

Предмет: програмно-апаратний комплекс для інтерактивної зовнішньої реклами .

Для того щоб досягти поставленої мети необхідно вирішити такі **завдання:**

- проаналізувати існуючі види реклами та виявити їх переваги та недоліки;
- пояснити вибір складових, компонентів та технологій для програмування;

-
- створити алгоритму пристрою зовнішньої реклами ;
 - розробити програмне забезпечення приладу для показу рекламного повідомлення ;
 - показати результат роботи готового апарату;
 - оцінити майбутню преспективу апарату та його допрацювань;
 - розробити питання з охорони праці та безпеки життєдіяльності.

Практичне значення отриманих результатів полягає у тому, що розроблений апаратний комплекс буде мати змогу надати певні новинки в зовнішній рекламі, таким чином сприяючи розвитку маркетингу. Завдяки характеристикам Arduino буде досягнуто високий рівень взаємодії, що дозволить людям насолоджуватись приємною графікою. Такий комплекс буде сприяти більшої здатності до виконання різних задач. Це одна з найпомітніших переваг, вона дає можливість ефективніше виконувати завдання поставлені маркетологом, а також виконувати багато функцій одночасно. Даний комплекс передбачає показ реклами без зайвих затримок, що може покращити здатність зацікавлювати інших людей та мати кращу взаємодію. Соціальний аспект реклами значно виріс за останні 10 років. Тепер все що цікавило людину може показуватись в вашому браузері. Також вже не є великою потребою клеїти бумажки з певним текстом , адже можна просто вивести цей текст на монітор й зацікавити більше людей ніж можливо було раніше. Зараз рекламний бізнес став дуже поширеним й йому потрібні певні удосконалення.

РОЗДІЛ 1

АНАЛІТИЧНИЙ ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ ТА СКЛАДОВИХ СИСТЕМИ

1.1 Реклама

Реклама - це рекламна діяльність, яка спрямована на продаж продукту або послуги цільовій аудиторії. Це одна з найстаріших форм маркетингу, яка намагається вплинути на дії своєї цільової аудиторії, щоб купити, продати або зробити щось конкретне. Використовуючи чітко адаптоване повідомлення, реклама може бути нішовою (націлена на невелику аудиторію) або загальною (орієнтована на велику аудиторію).

Реклама набагато старіша за більшість інших маркетингових заходів, таких як маркетинг електронною поштою та маркетинг у пошукових системах. Оскільки Інтернет став нормою, рекламу поділили на дві сфери: традиційну рекламу та цифрову рекламу.

Традиційна реклама стосується друкованої, теле- та радіореклами, яка користується популярністю понад 150 років. Друкована реклама є найефективнішою рекламою для бізнесу, оскільки вона обертається навколо цільової аудиторії, яка особисто отримує рекламу через листівки, газети та пошту.

Цифрова реклама ґрунтується на будь-якій рекламній діяльності в Інтернеті, як-от медійна реклама, PPC, реклама в соціальних мережах тощо. Ця форма реклами дешевша й легше відстежується, тому вона стала більш широко використовуваною формою маркетингу.

Рекламу (рис. 1.1) можна класифікувати за:

- принципом роботи;
- цільовою аудиторією ;
- універсальністю ;

- креативом;
- тривалістю показу;
- габарита ;
- вартістю показу оголошення.



Рисунок 1.1 – Різновиди реклами .

1.2 Аналіз основних характеристик конкурентів

Сітілайт

Це закрита конструкція, яку можна побачити вздовж тротуарів, а також поряд із торгови центра , зупинка та в інших людних місцях. Сітілайти не

бояться негативного впливу погодних умов, вони оснащені підсвічуванням і дозволяють розміщувати інформацію з обох боків.

Такий формат зовнішньої реклами орієнтований переважно на пішоходів. Він розташований на рівні очей, тому забезпечує більш тривалий візуальний контакт із клієнта. Це дозволяє розмістити на ньому більше інформації, ніж білборд.



Рисунок 1.2 – Сітілайт

Характеристики:

- формат TIF;
- роздільна здатність для друку – 75 dpi;
- розмір макета – 1200x1800 мм, робоче поле – 1100x1700 мм (фон робить навиліт);
- колірна модель – СМУК;
- шари зведені один;
- відсутність альфа-каналів;

- стиснення LZW (зменшує розмір файлу без втрати якості);
- превью у форматі JPEG чи PNG до 1 Мб.

У таблиці 1.1 наведені плюси і мінуси даного виду.

Таблиця 1.1 – Плюси та мінуси Сітілайт

Переваги	Недоліки
Простота застосування	Постійне місце дислокації
Зовнішня привабливість	Вартість розміщення
Можливість змінювати текст	Відсутність автономного живлення
Розповсюдженість	
Великий відсоток креативу	

Транзитна реклама:

Це реклама, яку розміщують на об'єктах транспортної інфраструктури, а також зовні та всередині автобусів, тролейбусів, автомобілів та інших транспортних засобів. Її величезною перевагою є широке охоплення. Наприклад, реклама, розміщена на бортах автобуса, щодня курсує вулиця міста та привертає увагу навіть тих, хто не пересувається громадським транспортом. Найчастіше такий формат використовують великі компанії із великим рекламним бюджетом.



Рисунок 1.3 – Транзитна реклама

Характеристики:

- застосування – на різних видах транспорту;
- живлення – не потребує;
- постійний показ одного повідомлення;

Таблиця 1.2 – Плюси та мінуси Транзитної реклами

Переваги	Недоліки
Одна з найдешевших	Відсутня інтерактивність
Привертає багато уваги за рахунок великих розмірів	Не можна змінювати зміст реклами
Різні носії	
Постійна зміна дислокацій	

Білборд:

Є великим щитом, який встановлюють на вулицях і вздовж трас. Білборди чудово працюють для привернення уваги як пішоходів, так і автомобілістів.

Великі рекламні щити складно не помітити, тому за їх допомогою можна легко та швидко охопити велику кількість цільової аудиторії.

Головне, продумати оформлення, оскільки потенційні клієнти переглядають його лише близько трьох секунд. У зв'язку з цим замість довгих текстів краще використовувати слогани і короткі фрази, що складаються приблизно з шести слів.



Рисунок 1.4 – Білборд

Характеристики:

- висота 3 метри та ширина 6 метрів;
- застосування – в приміщенні;
- фіксація – на металевому закріпленні ;

Таблиця 1.3 – Плюси та мінуси Білбордів

Переваги	Недоліки
Висока передача інформації за рахунок розміру	Вигорає на сонці
Зазвичай встановлені в людних місцях	Висока вартість
Можливість повторного показу	Не інтерактивний

Штендер:

Відносно невелика переносна конструкція, яку розміщують поруч із торговими точками, салона краси, кафе, ресторана та бара для привернення уваги перехожих. У порівнянні з іншими видами зовнішньої реклами штендери менш затратні. Вони прості в установці і легко можна їх переміщати з одного місця в інше. Найчастіше на вулицях зустрічаються прямокутні або арочні штендери, однак є і фігурні.



Рисунок 1.5 – Штендер

Характеристики:

- вага – 3 кг;
- рекламне поле – широкоформатний друк;
- двосторонній Т подібний;

Таблиця 1.4 – Плюси та мінуси Штендеру

Переваги	Недоліки
Маленька вага конструкції	При поганому нанесенні рекламний матеріал може тріснути
Міцність і стійкість до вітру	Потрібно постійно виносити та заносити
Доступність по ціні	

Тривалий термін служби

Компактні габарити

Жива реклама:

Це зовнішня реклама, розміщена людях. Такий спосіб донесення інформації до цільової аудиторії передбачає використання костюмів, плакатів-показчиків, табличок. Найчастіше живу рекламу можна побачити поруч із торговою точкою для залучення покупців. При правильному підході вона не тільки підвищує інтерес потенційних покупців, але й формує емоційний зв'язок, що сприяє покупці.



Рисунок 1.6 – Жива реклама

Таблиця 1.5 – Плюси та мінуси живої реклами

Переваги	Недоліки
Мобільність	Потрібна зацікавлена людина
Невисока вартість	
Можливий креатив	

1.3 Порівняння моделей-конкурентів

При виборі будь-якого виду реклами, необхідно спиратися на характеристики реклами, розуміючи що найбільш важливе саме для вас.

Головні критерія для аналізу були мобільність, креативність, можливість комунікувати з іншими, розміри рекламного оголошення, інтерактивність та напевно найважливіший критерій вартість.

Головні правила вибору певного виду реклами:

При виборі реклами потрібно звертати увагу на наступні моменти:

1. Актуальність. Вона повинна бути затребуваною;
2. Креативність. Адже нудна реклама не принесе ніякої користі;
3. Зрозумілість. Людина при перегляді повинна одразу розуміти про що йде мова в оголошенні;

Порівняльну характеристику розглянутих видів реклами наведено в таблиці 1.6.

Таблиця 1.6 – Порівняльна таблиця моделей-конкурентів

Вид реклами	Сітілайт	Транзитна реклама	Білборд	Штендер	Жива реклама
Характеристика					
Креативність	Середня	Висока	Висока	Низька	Висока
Мобільність	Відсутня	Висока	Відсутня	Середня	Висока
Можливість комунікувати з іншими	Відсутня	Відсутня	Відсутня	Відсутня	Можливо
Розміри рекламного оголошення(м)	1.8*1.2	В залежності від розміру транспортного засобу	6*3	0.6*1	В залежності від розміру людини
Інтерактивність	Присутня	Відсутня	Відсутня	Відсутня	Відсутня
Вартість	Від 2900 гривень	Від 4000гривень	Від 6600 гривень	Від 1800 гривень	Від 1000 гривень

Порівнявши дані види реклами , можемо зробити висновки, що сітілайт є найоптимальний варіант, при великій кількості потрібних характеристик має доволі прийнятну ціну. Тому якщо необхідна велика зацікавленість споживачів, то потрібно вибирати білборд хоча вартість є високою.

Тому прийнято рішення розробити прилад для показу реклами , який вбирає в себе тільки важливі складові, та буде відповідати прийнятній ціні.

1.4 Вибір комплектуючих (апаратна база)

Щоб почати збирати апаратний комплекс спочатку необхідно визначитись , якій ціні та характеристикам повині відповідати комплектуючі.

По закінченю порівняння конкурентів,потрібно обрати найкращий варіант з усіма необхідни перевага та характеристика для апарату.

Для програмно-апаратного комплексу необхідні:мкроконтролер та матричний дисплей.

1.4.1 Мікроконтролер

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 дошка невеликого розміру, а також 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. Серед інших плат розробника: AVR Development Board, PIC Development Board, Raspberry Pi , Intel Edison, MSP430 Launchpad та ESP32

Arduino NANO — це плата мікроконтролера, широко використовується студента електроніки для різних проектів через менший розмір порівняно з інши плата .

Ця плата має численні можливості та особливості, як плата Arduino Duemilanove. Як би там не було, ця плата Nano різноманітна в комплектації.

Він не має жодного роз'єму постійного струму, щоб керувати живленням, використовуючи порт USB, щось інше, безпосередньо пов'язане з контактами, такі як VCC і GND. Ця плата може бути забезпечена напругою від 6 до 20 вольт, використовуючи менший, ніж очікувалося, порт USB на платі.

Основні відмінності між Uno та Nano перелічені у таблиці 1.7.

Таблиця 1.7 – Порівняльна таблиця Arduino UNO та Arduino Nano

Specification	Arduino Uno	Arduino Nano
Processor	ATmega 328P	ATmega 328P
Input Voltage	5V/7-12V	5V/7-12V
Speed of CPU	16 MHz	16MHz
Analog I/O	6/0	8/0
Digital IO/PWM	14/6	14/6
EEPROM/SRAM [kB]	1/2	1/2
Flash	32	32
USB	Regular	Mini
USART	1	1

Я віддав перевагу Arduino Uno (рис. 1.7).

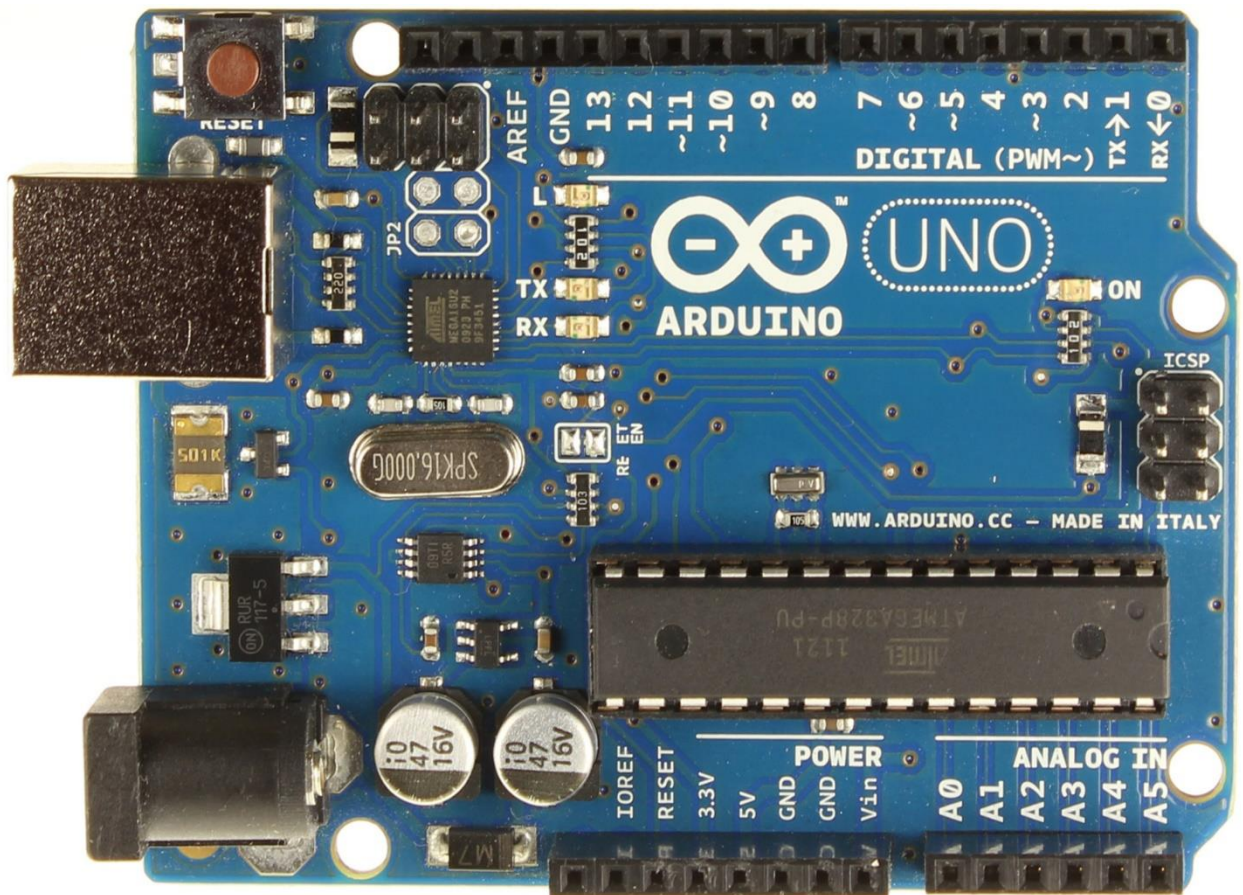


Рисунок 1.7 – Arduino UNO

Плата Arduino Nano порівнянна з порівняльним мікроконтролером Arduino UNO, таким як Atmega328p. Таким чином вони можуть ділитися порівнянною програмою. Найбільша відмінність між цими двома — міра. Оскільки оцінка Arduino Uno подвійна до плати NANO. Таким чином, листи Uno використовують більше місця на каркасі. Якщо подається більше напруги або струму, ніж рекомендований, є висока ймовірність спалити чіп, і його можна легко замінити в UNO, тоді як вже не можна повернути назад пошкодження в NANO.

1.4.2 Світлодіодна матриця

Для даного проекту обрано світлодіодну матрицю MAX7219 (рис. 1.8).

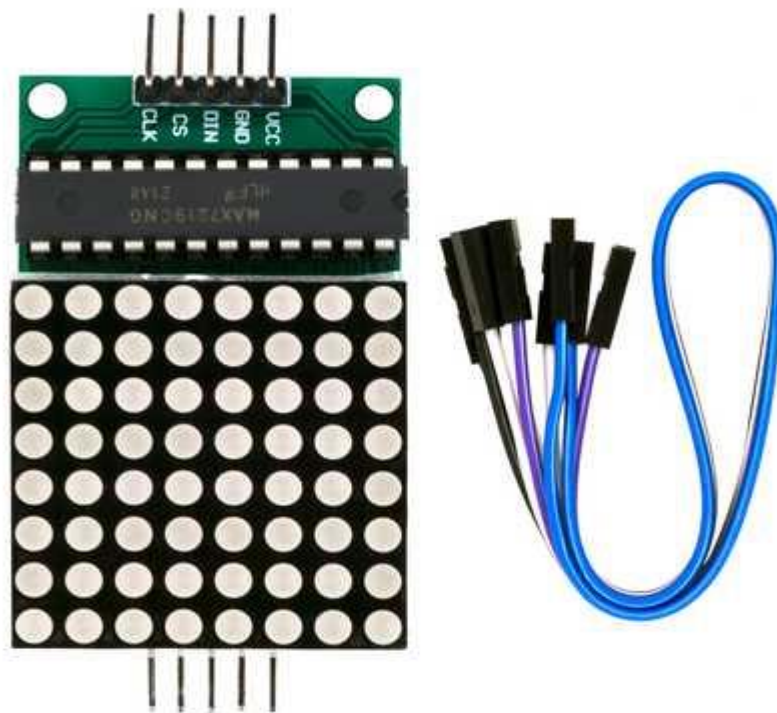


Рисунок 1.8 – Світлодіодну матрицю MAX7219

Світлодіодна матриця — це двовимірний масив світлодіодів, які можна використовувати для відображення символів, символів або навіть зображень. Залежно від орієнтації світлодіодів у матриці, світлодіодні матриці можуть бути двох типів. Це анод загального ряду та катод загального ряду. Світлодіодні матричні модулі є одні з часто використовуваних пристроїв відображення і використовуються в основних програмах, таких як електронні панелі відображення та систе сповіщень.

У проекті для відображення інформації використовується світлодіодний матричний дисплей 8 x 8. Світлодіодні матриці доступні в різних стилях, таких як одноколірна, двоколірна, багатоколірна або світлодіодна матриця RGB. Вони також доступні в різних розмірах, таких як 5 x 7, 8 x 8, 16 x 16, 32 x 32 тощо. Залежно від розташування світлодіодів у матриці, світлодіодна матриця може бути або анодом із звичайним рядком, або катодом із звичайним рядом.

У разі світлодіодної матриці загального рядкового анодного типу джерела струму (висока або позитивна напруга) подаються в рядки AD, а поглиначі струму (низька або негативна напруга або земля) – у стовпці 1-4.

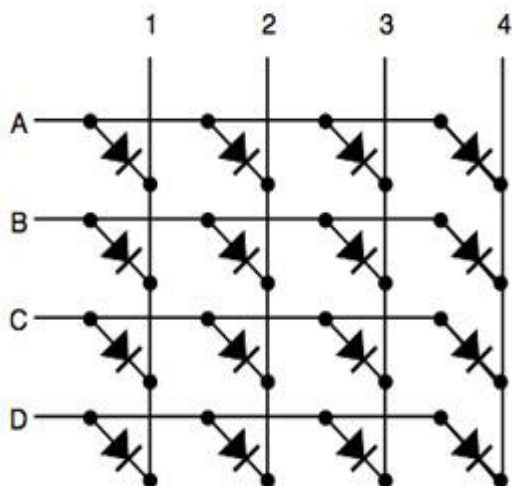


Рисунок 1.9 – Світлодіодної матриці загального рядкового анодного типу

У разі світлодіодної матриці катодного типу загального ряду джерела струму (висока або позитивна напруга) подаються в стовпці 1-4, а поглиначі струму (низька або негативна напруга або маса) – у рядки AD.

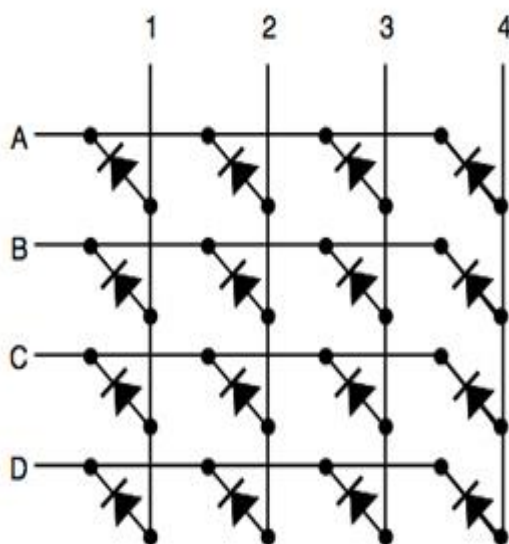


Рисунок 1.10 – Світлодіодної матриці катодного типу

Світлодіодна матриця, яка використовується в проекті, являє собою звичайну світлодіодну матрицю катодного типу. При розробці проекту необхідно знати тип світлодіодної матриці і відповідно до цього написати програму.

MAX 7219 — звичайний катодний драйвер дисплея з послідовним входом і паралельним виходом. Він використовується для взаємодії між мікропроцесора та мікроконтролера з 64 окре світлодіода (наприклад, матриця 8 x 8 світлодіодів має 64 світлодіоди), се сегментни світлодіодни дисплея до 8 цифр або індикатора з гістограмою.

Світлодіодна матриця 8 x 8 підключається до MAX 7219, як показано на принциповій схемі, і вхідні дані надходять з плати Arduino на MAX 7219.

Технічні характеристики MAX7219:

- кількість індикаторів : 4;
- тип індикаторів: загальний катод;
- розмір одного індикатора: 32*32*8 мм;
- кількість світлодіодів на індикаторі: 64;
- діаметр одного світлодіода: 3 мм;
- колір світлодіода: червоний;
- інтерфейс підключення матриці:SPI;
- напруга живлення: 5 В;
- розміри матриці: 130*32*14мм.

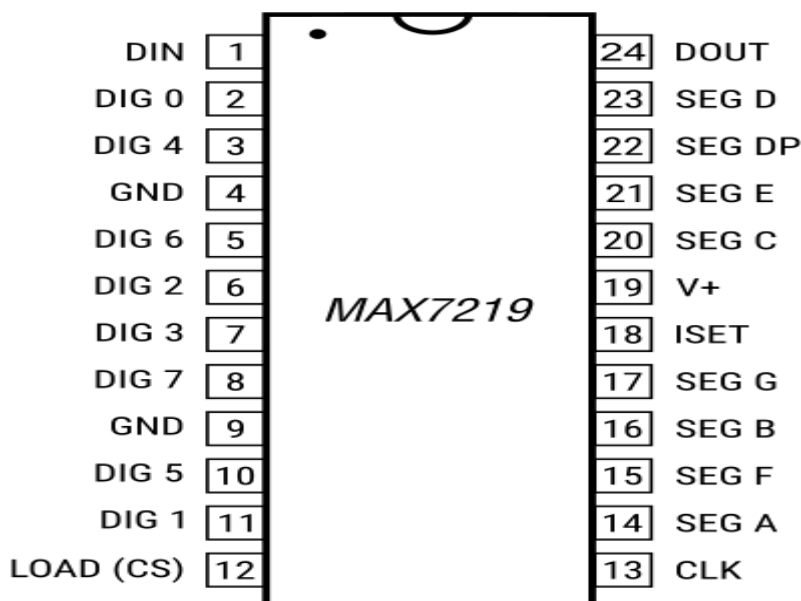


Рисунок 1.11 –Розпинівку мікросхе MAX7219

Висновки до розділу 1

Було проаналізовано ключові характеристики аналогів та конкурентів на ринку та виділено їх недоліки. Встановлено, що сітілайт є хорошим варіантом для зовнішньої реклами, при великій кількості переваг та характеристик має допустиму ціну. Але якщо необхідна максимальна креативність та увага то білборд буде кращим, хоча й має великий недолік в вартості та відсутність інтерактивності.

Оптимальний набір для подальшого розроблення складається з мікроконтролера.

Провів аналіз різних видів реклами та поділив їх за класифікацією.

Після порівняльного огляду, відповідно до потрібних характеристик та прийнятній ціні, обрано комплектуючі: мікроконтролер Arduino Uno та Світлодіодну матрицю.

РОЗДІЛ 2

РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ

2.1 Детальний огляд світлодіодної матриці

Це масив світлодіодів, розташованих у рядках і стовпцях, які можна використовувати для відображення цифр, букв і фігур. Вони бувають різних розмірів, але найпоширенішою є світлодіодна матриця 8×8, тобто 64 світлодіоди, розташовані у 8 рядках і 8 стовпцях.

Світлодіоди підключаються як із загальним анодом, так і з загальним катодом. В обох випадках всі аноди світлодіодів у ряду або стовпці з'єднані разом, і всі катоди світлодіодів у стовпці або рядку також з'єднані разом.

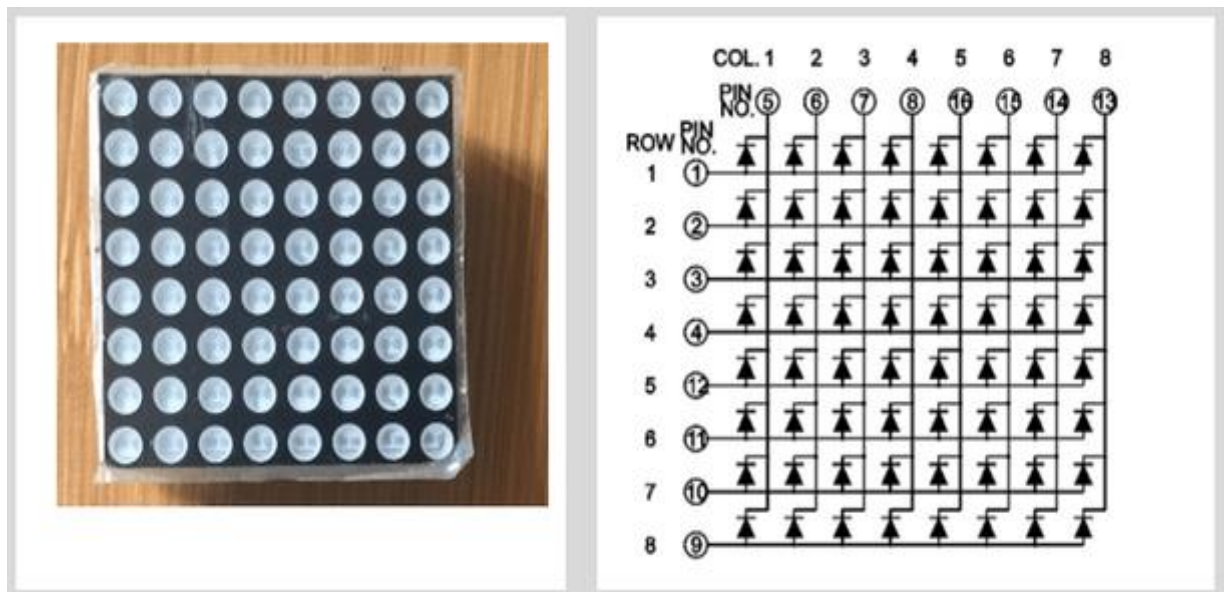


Рисунок 2.1– Розміщення світлодіодів

2.1.1 Як працює світлодіодна матриця

Неможливо ввімкнути декілька світлодіодів у різних рядках і стовпцях і утримувати їх увімкненими одночасно. Для створення візуального ефекту, коли всі призначені світлодіоди здаються ввімкненими одночасно, використовується метод **збереження зору**.

Людські очі запам'ятовують спалах світла приблизно на 20 мс, тому, коли людина постійно в кає світлом із швидкістю 20 мс або швидше, здається, що світло ніколи не згасає.

Використовуючи цю концепцію, мікроконтролер в кає необхідні світлодіоди в рядку або стовпці, один рядок або один стовпець за іншим, перебираючи всі рядки або стовпці, щоб увімкнути потрібні світлодіоди, зі швидкістю більше 20 мс, що вводить людські очі в обман. побачивши, що призначені світлодіоди одночасно горять!

2.1.2 Використання мікросхе МАХ7219 для керування світлодіодною матрицею 8×8.

Щоб керувати світлодіодною матрицею 8×8 безпосередньо за допомогою мікроконтролера, потрібно підключити кожен рядок і кожен стовпець до цифрового контакту, що означає, що знадобиться 16 контактів вводу/виводу. Це занадто багато контактів для керування одним пристроєм.

Щоб вирішити цю проблему, потрібно використовувати мікросхему МАХ7219 для керування 64 світлодіода , використовуючи лише 3 дроти для підключення до мікроконтролера. Крім того, можливо навіть поєднати кілька мікросхем МАХ7219 для великих дисплеїв без необхідності в додаткових контактах вводу/виводу.

2.1.3 МАХ7219 Конфігурація контактів

МАХ7219 — це 24-контактна мікросхема, розроблена як драйвер дисплея із послідовним введенням із загальним катодом, що використовується для взаємодії між мікроконтролера з 7-сегментни цифрови світлодіодни дисплея до 8 цифр, дисплея з гістограмою або 64 окре світлодіода . У цьому випадку його використовують для керування світлодіодною матрицею 8×8.

Наведено схему та опис кожного контакту мікросхе МАХ7219.

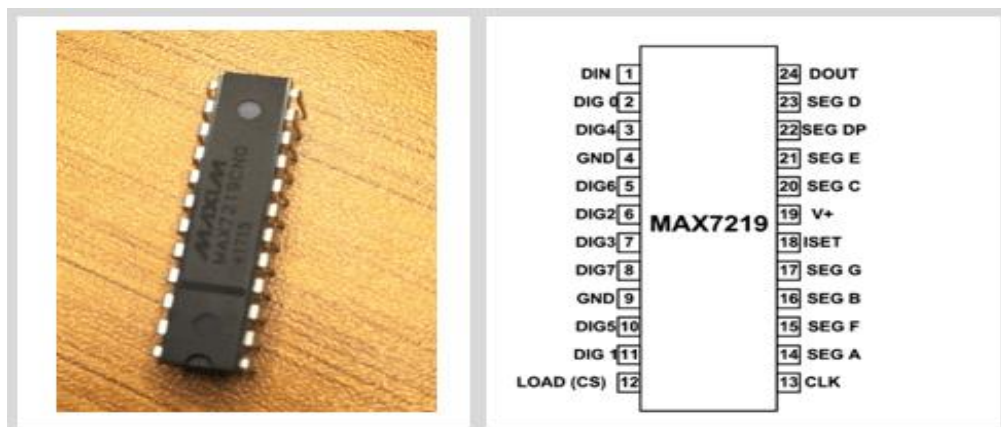


Рисунок 2.2– Опис контакту мікросхе МАХ7219

- **DIG0 – DIG7:** загальний катод сегментів дисплея, тобто всі вісім сегментів від цифр 0 до цифри 7 мають спільну землю.
- **SEG A – SEG G:** загальний термінал сегмента для всіх восьми цифр.

Функціональні контакти:

- **DIN** – Послідовний контакт для введення даних.
- **LOAD(CS)** – Вибір мікросхеми або контакт зсуву даних.
- **CLK** – Годинник.
- **DOUT** – Контакт використовується для послідовного підключення другого мікросхеми.
- **ISET** – штифт регулювання виходу струму.

2.1.4 Підключення мікросхеми MAX7219 до світлодіодної матриці

На рисунку 2.3 показано, як MAX7219 підключається до світлодіодної матриці 8×8, а також як різні компоненти розташовані на комерційних комутаційних платах.

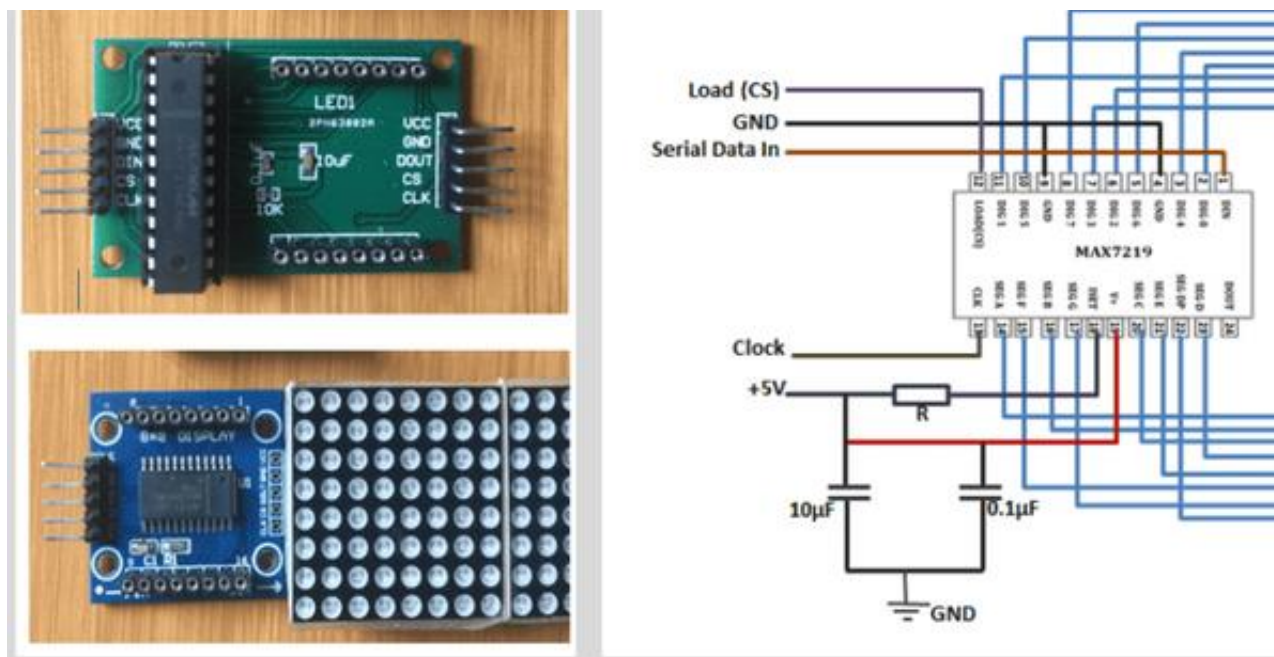


Рисунок 2.3–Підключення MAX7219 до матриці

Вивід MAX7219 з позначкою SEG A~F підключені до висновків рядка світлодіодної матриці, а висновки DIG 0~7 до стовпців відповідно.

Між 5 В і контактом 18 MAX7219 є резистор. Оскільки ця мікросхема є драйвером світлодіодів постійного струму, значення резистора використовується для встановлення потоку струму до світлодіодів, що важливо для контролю яскравості світлодіодів. світлодіоди.

Якщо ви розробити свій власний матричний світлодіод з MAX7219 IC, щоб вибрати правильний резистор, потрібно знати напругу і прямий струм для світлодіодної матриці, а потім зіставити значення зі значення в Rset vs Segment Current і LED таблиця прямої напруги з листа даних MAX7219, як показано нижче;

Iseg (mA)	Vled (V)				
	.5	.0	.5	.0	.5
40	2.2	1.8	1.0	0.6	.69
30	7.8	7.1	5.8	5.0	4.0
20	9.8	8.0	5.9	4.5	2.6
10	6.7	3.7	9.3	5.4	1.2

Якщо є світлодіод 2 В 20 мА, значення резистора буде 28 кОм

2.1.5 Підключення світлодіодної матриці MAX7219 до Arduino

Драйвер MAX7219 зв'язується з Arduino через SPI (послідовний периферійний інтерфейс), де Arduino діє як головний пристрій, а світлодіодна матриця буде веденою.

Щоб досягти цього, знадобляться лише три підключення:

- **Лінія даних (DIN)** – це MOSI (Master Out Slave In) для зв'язку SPI
- **Serial Clock (CLK)** – тактовий імпульс, який синхронізує передачу даних, згенерованих провідним.
- **Вибір мікросхе (CS)** – PIN-код на кожному пристрої, який головний може використовувати для ввімкнення та вимкнення певних пристроїв.

Зв'язок SPI може базуватися на апаратному або програмному забезпеченні. Апаратний SPI швидший, але працює лише з конкретни висновка SPI для даної плати Arduino, наприклад, для Arduino UNO контакти SPI мають 10, 11 і 13 для вибору мікросхе (CS), даних (DIN) і тактової частоти (CLK) відповідно.

При використанні програмного SPI ви можете використовувати будь-який цифровий pin Arduino, але потрібно оголосити ці контакти в коді.

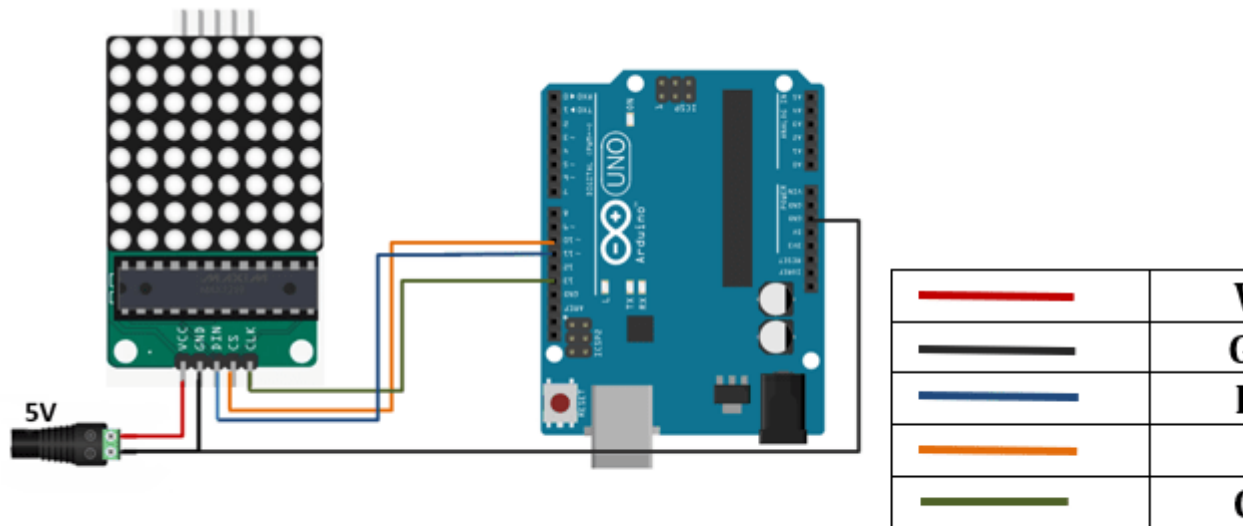


Рисунок 2.4 – Підключення Arduino

Однією з головних переваг чіпа MAX7219 є те, що він дає змогу об'єднати декілька світлодіодних матриць 8×8 для створення більшого дисплея, який все ще використовує лише 3 контакти для зв'язку SPI. Щоб об'єднати матриці послідовно, з'єднайте DOUT першого дисплея з DIN наступного дисплея, а інші контакти, тобто VCC, GND, CLK і CS будуть спільні між дисплея .

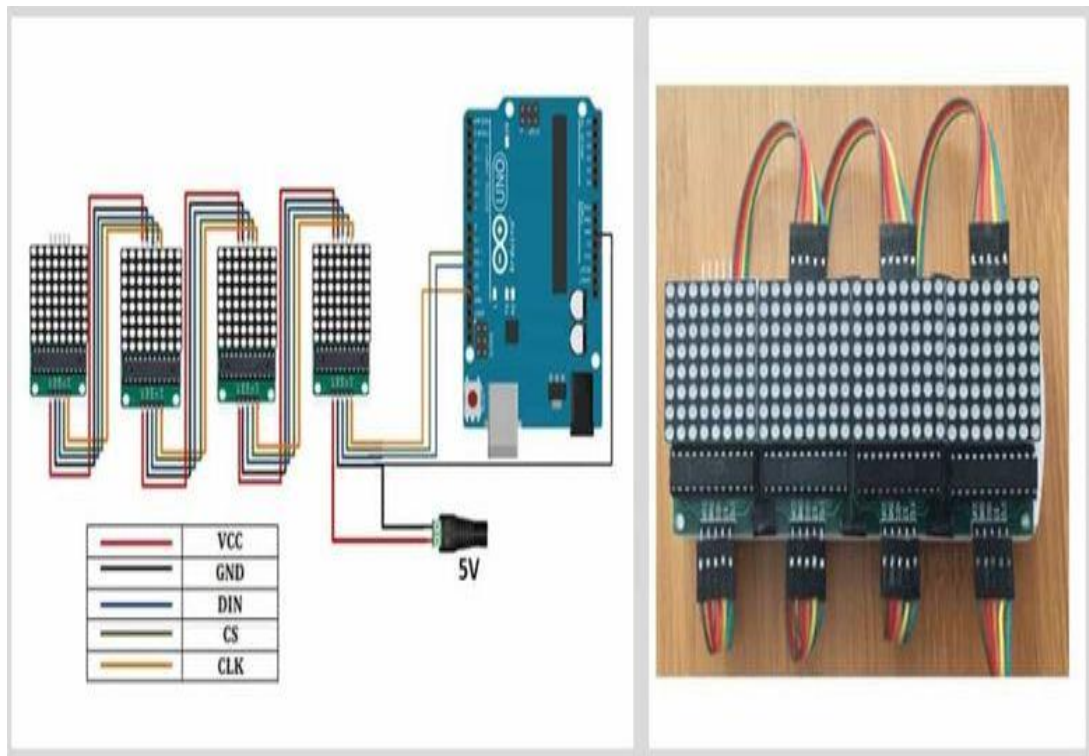


Рисунок 2.5– Об'єднання декілька світлодіодних матриць

Модуль FC-16 вже має з'єднання для послідовного з'єднання світлодіодних матриць 8×8, зроблених на платі розподілу, тому потрібно лише підключити вихідні контакти VCC, GND, DIN, CLK і CS до Arduino, як показано на рисунку нижче.

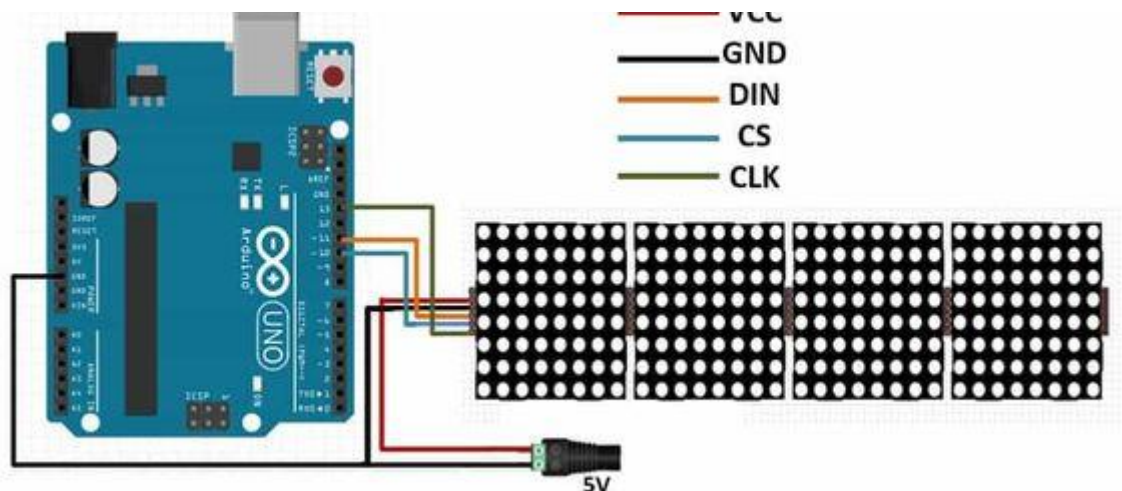


Рисунок 2.6– Об'єднання декілька світлодіодних матриць схематично

2.1.6 Про потужність

Максимальна потужність, яку може безпечно забезпечити Arduino Uno при живленні від USB, становить близько 400 мА при напрузі 5 В.

Світлодіодні матричні дисплеї мають досить високе споживання струму і можуть використовувати до 1 А, якщо яскравість встановлена на максимум.

Завжди використовується зовнішнє джерело живлення для дисплея. Ніколи не підключаються більше чотирьох світлодіодних матриць безпосередньо до мікроконтролера і завжди тримається яскравість менше 50%, щоб уникнути руйнування регулятора напруги мікроконтролера.

2.1.7 Керування світлодіодною матрицею 8×8 за допомогою MAX7219 та Arduino

Є два способи використання мікросхеми MAX7219 для керування світлодіодною матрицею. Один з них використовує процедурне програмування для передачі даних побітно, а другий спосіб полягає у використанні раніше написаних бібліотек для цього чіпа.

Використання бібліотек є найпростішим способом отримати необхідний результат, оскільки немає необхідності турбуватися про детальне програмування. Потрібно лише ввести необхідні дані для відображення на світлодіодній матриці. Нижче буде продемонстровано, як використовувати обидва методи в прикладах ;

Драйвер світлодіода MAX7219:

- Заощаджує **контакти мікроконтролера** ..
- Заощаджує час **обробки** !
- Управляє 64 світлодіода (або більше) за допомогою 3-контактного інтерфейсу,
- ...Або керує 8 х **се сегментни** дисплея .

Нижче можливо точно дізнатися, чому це правда і як можливо використовувати ці пристрої на Arduino. Також можна побачити, як використовувати кілька пристроїв без використання додаткових контактів процесора!

Використовуючи 7219, можете керувати 64 світлодіода, тоді як потрібно лише 3 дроти для підключення його до мікроконтролера (це за винятком V_{CC} і GND, які вважаються доступними). Крім того, можна об'єднати кілька мікросхем 7219 для більших дисплеїв.

Є 16 вихідних ліній від 7219, які управляють 64 індивідуальні світлодіода . Це звучить неможливо, але метод водіння використовує те, як працюють людські очі. Постійність зору використовується для того, щоб світлодіоди здавалися включени весь час, хоча насправді це не так. Фактично світлодіоди розташовані як набір рядків і стовпців 8x8. Кожен стовпець пульсує протягом короткого часу, поки обробляються біти рядка для цього стовпця.

Людські очі запам'ятовують спалах світла приблизно за 20 мс, тому, коли ви постійно спалахуєте світлом (або світлодіодом) зі швидкістю 20 мс або швидше, здається, що світло ніколи не згасає. Ось як працює 7219. Усі світлодіоди окремо в каються на короткий час із частотою більше 20 мс.

2.1.8 Інтерфейс SPI та Специфікація MAX7219

MAX7219 має чотирипровідний інтерфейс SPI - годинник, дані, вибір

1. Дані - **MOSI** - головний вихід, послідовний вхід. 7219 є підпорядкованим пристроєм.
2. Вибір мікросхе - **Завантаження (CSn)** - активний низький вибір мікросхе .
3. Годинник - **SCK**
4. **Земля.**

Параметр	Значення
Напруга живлення (Vs)	4,0 В ~ 5,5 В
Струм живлення	330 мА
Сегмент струму джерела приводу	-40 мА
Вимкнути струм	150 мкА
Швидкість сканування	500-1300 Гц, 800 Гц (тип.)
Інтерфейс	<u>SPI</u> (чотири дроти)

Максимальний послідовний годинник	10 МГц (макс.)
Робоча температура (MAX7219E)	-40°C ~ 85°C

2.2 З'єднання всієї світлодіодної матриці разом

Кожен 3-контактний роз'єм на схемі вище символізує один модуль, описаний у попередньому розділі (LED Matrix + MAX72xx), тепер з'єднали всі ці модулі разом.

Усі мікросхеми MAX72xx мають спільні лінії MOSI та SCK, MISO не використовується, кожен чіп займає окрему лінію Slave Select.

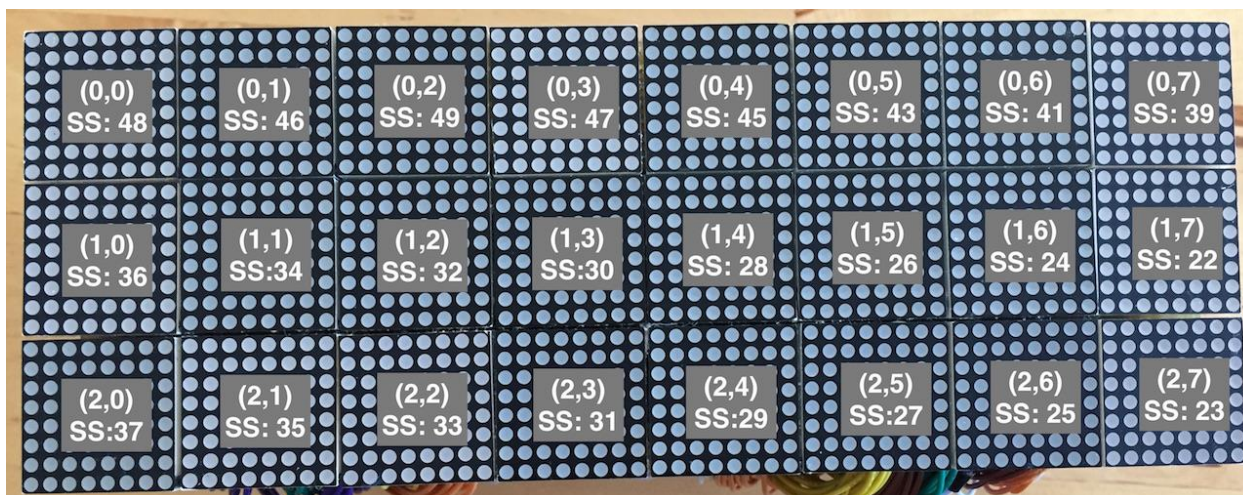


Рисунок 2.7–Підключення світлодіодних матриць

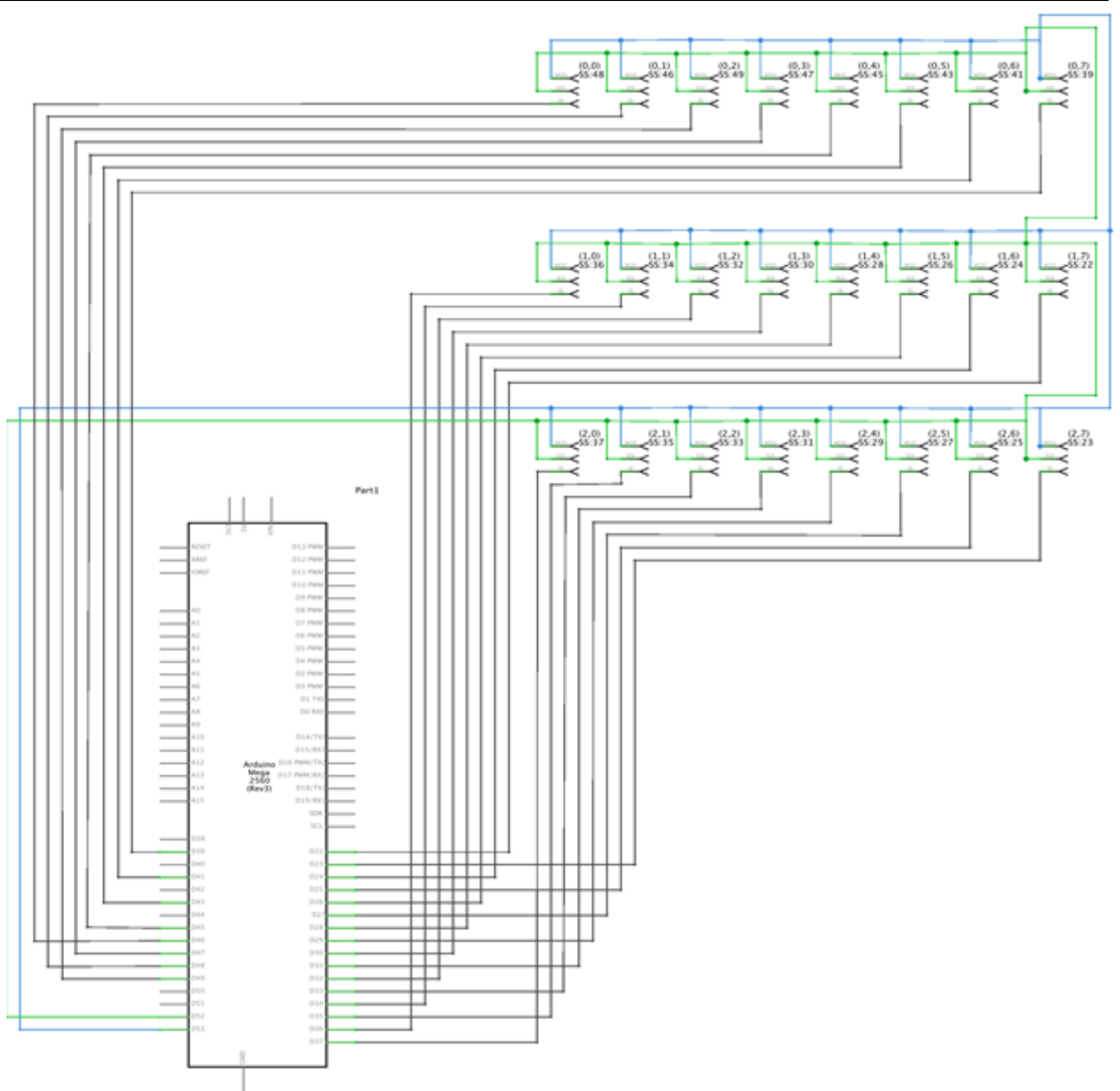


Рисунок 2.8–Схематичне підключення елементів світлодіодних матриць

Положення світлодіодної матриці на схемі вище безпосередньо відповідає їх розташуванню на фізичному дисплеї, який я використовував для тестування. Крім того, кожен модуль має опис, що вказує його позицію та рядок Select Slave, наприклад: (2,1) SS: 35 дає нам другий модуль у третьому рядку (відлік від нуля) і PIN:35 на Arduino для лінії Select Slave.

2.3 Детальний огляд Arduino Uno

Arduino UNO має максимальний номінальний струм 40 мА , тому навантаження не повинно перевищувати цей номінальний струм, інакше можна пошкодити плату.

Він поставляється з кварцевим генератором 16 МГц, що є його робочою частотою.

Розпинівка Arduino Uno складається з 14 цифрових контактів, починаючи з D0 до D13 .

Має 6 аналогових контактів, починаючи від A0 до A5 .

Має 1 контакт для скидання , який використовується для програмного скидання плати. Для того, щоб скинути плату, нам потрібно зробити цей контакт LOW.

Має 6 контактів живлення , які забезпечують різні рівні напруги.

З 14 цифрових контактів 6 контактів використовуються для генерації імпульсів ШІМ з роздільною здатністю 8 біт. ШІМ-контакти в Arduino UNO - це D3, D5, D6, D9, D10 і D11 .

Arduino UNO постачається з 3 типа пам'яті , пов'язани з нею, під назвою:

Флеш-пам'ять: 32 КБ

SRAM: 2 КБ

EEPROM: 1 КБ

Arduino UNO підтримує 3 типи протоколів зв'язку , які використовуються для взаємодії з периферійни пристроя сторонніх розробників, під назвою:

Послідовний протокол

Протокол I2C

Протокол SPI

Крім USB, для живлення плати також можна використовувати акумулятор або адаптер змінного та постійного струму.

Особливості Плати Arduino Uno

Arduino Uno поставляється з інтерфейсом USB, тобто USB-порт додається на платі для розвитку послідовного зв'язку з комп'ютером.

Мікроконтролер Atmega328 розміщений на платі, яка має ряд функцій, таких як таймери, лічильники, переривання, ШІМ, центральний процесор, контакти вводу-виводу і заснований на тактовій частоті 16 МГц, що допомагає виробляти більшу частоту та кількість інструкцій за цикл.

Це платформа з відкритим вихідним кодом, де кожен може змінювати та оптимізувати плату на основі кількості інструкцій та завдань, які він хоче виконати.

Ця плата має вбудовану функцію регулювання, яка тримає напругу під контролем, коли пристрій підключено до зовнішнього пристрою.

На платі присутній пин-код для скидання, який скидає всю плату і запускає програму на початковому етапі. Ця шпилька корисна, коли плата зависає в середині запущеної програми; натискання цієї шпильки очистить все в програмі та запустить програму з самого початку.

На платі вбудовано 14 цифрових і 6 аналогових контактів, що дозволяє підключати до будь-якої схеми на платі. Ці контакти забезпечують гнучкість і простоту використання зовнішніх пристроїв, які можна підключити за допомогою цих контактів. Для підключення пристроїв до плати не потрібен жорсткий і швидкий інтерфейс. Просто підключіть зовнішній пристрій до контактів плати, які розміщені на платі у вигляді заголовка.

6 аналогових контактів позначені від A0 до A5 і мають роздільну здатність 10 біт. Ці контакти вимірюють від 0 до 5 В, однак їх можна налаштувати на високий діапазон за допомогою функції `analogReference()` і контакту AREF.

Щоб увімкнути плату, потрібно лише 5 В, що можна досягти безпосередньо за допомогою порту USB або зовнішнього адаптера, однак він може підтримувати зовнішнє джерело живлення до 12 В, яке можна регулювати та обмежувати до 5 В або 3,3 В на основі вимоги проекту.

Arduino Uno заснований на мікроконтролері AVR під назвою Atmega328. Цей контролер має 2 КБ SRAM, 32 КБ флеш-пам'яті, 1 КБ EEPROM. Плата Arduino має 14 цифрових і 6 аналогових контактів. Вбудований АЦП використовується для вибірки цих контактів. На платі встановлений кварцевий генератор з частотою 16 МГц. На наступному малюнку показано розпинівку плати Arduino Uno.

На платі розміщено кілька цифрових і аналогових контактів вводу-виводу, яка працює при напрузі 5 В. Ці контакти мають стандартні робочі характеристики від 20 мА до 40 мА. У платі використовуються внутрішні підтягуючі резистори, які обмежують струм, що перевищує задані умови роботи. Однак занадто велике збільшення струму робить ці резистори марні і пошкоджує пристрій.

Світлодіодний режим. Arduino Uno постачається з вбудованим світлодіодом, який підключається через контакт 13. Якщо надати високе значення на контакт, він увімкнеться, а LOW – вимкнеться.

Це вхідна напруга, яка подається на плату Arduino. Він відрізняється від 5 В, що подається через порт USB. Цей контакт використовується для подачі напруги. Якщо напруга подається через роз'єм живлення, до нього можна отримати доступ через цей контакт.

Плата оснащена можливістю регулювання напруги. Контакт 5 В використовується для забезпечення вихідної регульованої напруги. Живлення плати здійснюється трьома способами, тобто через USB, Vin-контакт плати або роз'єм живлення постійного струму.

USB підтримує напругу близько 5 В, тоді як Vin і Power Jack підтримують напругу в діапазоні від 7 В до 20 В. Рекомендується експлуатувати плату від 5В. Важливо відзначити, що якщо напруга подається через контакти 5 В або 3,3 В, вони призводять до обходу регулятора напруги, що може пошкодити плату, якщо напруга перевищить її межу.

GND. Це заземлювачі. На платі передбачено більше одного контактів заземлення, які можна використовувати відповідно до вимог.

Скинути. Цей штифт вбудований на плату, який скидає програму, що працює на платі. Замість фізичного скидання на платі, IDE поставляється з функцією скидання плати за допомогою програмування.

IOREF. Цей контакт дуже корисний для забезпечення опорної напруги на платі. Для зчитування напруги на цьому контакті використовується екран, який потім вибирає відповідне джерело живлення.

ШІМ. ШІМ забезпечується 3,5,6,9,10, 11 контактами. Ці контакти налаштовані на 8-бітний вихід ШІМ.

SPI. Він відомий як послідовний периферійний інтерфейс. Чотири контакти 10(SS), 11(MOSI), 12(MISO), 13(SCK) забезпечують зв'язок SPI за допомогою бібліотеки SPI.

AREF. Він називається аналоговим опорним. Цей контакт використовується для подачі опорної напруги на аналогові входи.

TWI. Він називається двопровідним інтерфейсом. Доступ до зв'язку TWI здійснюється через Wire Library. Для цього використовуються шпильки A4 і A5.

Послідовний зв'язок здійснюється через два контакти, які називаються Pin 0 (Rx) і Pin 1 (Tx).

Контакт Rx використовується для отримання даних, а контакт Tx використовується для передачі даних.

Зовнішні переривання. Контакти 2 і 3 використовуються для забезпечення зовнішніх переривань. Переривання викликається шляхом надання LOW або зміни значення.

Arduino Uno поставляється з можливістю взаємодії з іншою платою, мікроконтролера та комп'ютера Arduino. Atmega328, розміщений на платі, забезпечує послідовний зв'язок за допомогою контактів, таких як Rx і Tx. Atmega16U2, вбудований на плату, забезпечує шлях для послідовного зв'язку за допомогою драйверів USB com. Послідовний монітор надається в програмному забезпеченні IDE, яке використовується для надсилання або отримання текстових даних з плати. Якщо світлодіоди, розміщені на

контактах Rx і Tx, будуть блимати, це вказує на передачу даних. Arduino Uno програмується за допомогою програмного забезпечення Arduino, яке є кросплатформним додатком під назвою IDE, написаним на Java. Мікроконтролер AVR Atmega328, розміщений на базі, має вбудований завантажувач, який позбавляє вас від використання окремого записувача для завантаження програми на плату. Arduino Uno поставляється з широким спектром застосувань. Більша кількість людей використовує плати Arduino для розробки датчиків та інструментів, які використовуються в наукових дослідженнях. Нижче наведено кілька основних застосувань плати:

- Вбудована система.
- Система безпеки та оборони.
- Цифрова електроніка та робототехніка.
- Прилавок для паркування.
- Вагові машини.
- Таймер зворотного відліку світлофора.
- Медичний інструмент.
- Аварійне світло для залізниць.
- Домашня автоматизація.
- Про слова автоматизація.

Arduino має велику спільноту, яка розвиває та ділиться знання з широким колом аудиторії. Доступна швидка підтримка щодо технічних аспектів будь-якого електронного проекту. Коли ви вибираєте плату Arduino перед іншим контролером, вам не потрібно влаштовувати додаткові периферійні пристрої та пристрої, оскільки більшість функцій легко доступні на платі, що робить ваш проект економічним за своєю природою та вільним від великої кількості технічних знань.

2.4 Порівняння Arduino та Raspberry Pi

Raspberry Pi має багато різних моделей, які працюють від процесора ARM. Від оригінальної одноядерної моделі з тактовою частотою 700 МГц у 2012 році до сучасної чотириядерної моделі 1,5 ГГц. Моделі Arduino зазвичай живляться від мікроконтролерів Atmel і часто мають частоту менше 100 МГц. Наприклад, Arduino Uno працює на частоті 16 МГц.

Ці чіпи значно повільніші, ніж ті, що є в Raspberry Pi, але Arduino не має стільки накладних витрат, як запуск операційної системи Linux. Якби проводили тест на основі чистої «кінської сили», Raspberry Pi однозначно виграв би. Але є що сказати про процесор з фіксованою швидкістю. Він надійний і не має масштабування, яке може спричинити проблеми з часом для проектів, які потребують абсолютної точності.

За потужність процесора Raspberry Pi завжди був явним переможцем. Arduino Portenta H7 — потужна плата, але вона не може зрівнятися з Pi за потужністю. Найнижча специфікація Raspberry Pi, яку можемо придбати, — це Raspberry Pi Zero W, який має один процесор з тактовою частотою 1 ГГц і все ще забезпечує більшу потужність, ніж двоядерний 480 МГц STM32H747. Але Raspberry Pi потребує більшої потужності, оскільки він також працює під керуванням операційної системи.

Вимоги до живлення: Raspberry Pi проти Arduino

Коли справа доходить до споживання електроенергії, Raspberry Pi 4 досить голодна плата. У Raspberry Pi 4 був новий роз'єм живлення USB C і офіційний блок живлення з більш високим рейтингом, який забезпечував до 3 А для Pi та будь-яких підключених до нього пристроїв. Таким чином, Raspberry Pi 4 теоретично може працювати з потужністю до 15 Вт. Arduino Uno може витягнути максимум 500 мА через USB. Він може споживати більше струму, якщо використовується

Гнучкість тут є перевагою. Arduino Uno може працювати з діапазоном напруг, які регулюються до 5 В, необхідних для плати. Живлення може

подаватись через порт USB, роз'єм постійного струму (від 6 до 20 В, який направляється через стабілізатор 5 В) або через контакт VIN, який підключається безпосередньо до мікроконтролера, тому перед підключенням завжди перевіряйте правильну напругу. Споживання струму для Arduino Uno, на якому працює ескіз «блیمانья», становить близько 40 мА, але додавання компонентів збільшить кількість використовуваного струму.

Порівняння GPIO: Raspberry Pi проти Arduino

Виводи загального призначення (GPIO) є з'єднання з «мозком» плати, і за допомогою них код можна використовувати для взаємодії з навколишнім світом. Виводи GPIO є двостороннім зв'язком, вони можуть бути входа або вихода, і вони можуть використовувати спеціальні протоколи зв'язку.

GPIO Raspberry Pi складається з 40 контактів (120 при використанні обчислювального модуля), а контакти є сумішшю цифрових широтно-імпульсної модуляції (PWM) і спеціальних протоколів, таких як I2C, SPI і UART. Багато контактів GPIO використовуються для більш ніж однієї функції/протоколу.

Arduino Uno має менше контактів GPIO (Arduino Mega має набагато більше контактів GPIO), але він має основи, цифрові контакти, ШІМ, I2C, SPI. У Arduino є те, чого немає у Raspberry Pi, аналогові входи, які використовують постійний сигнал, як правило, напругу, як засіб передачі даних.

Можливо, йому не вистачає великої кількості контактів GPIO, але в Arduino є все, що нам потрібно, щоб почати з електронних проектів, включаючи аналогові входи. Це може здатися невеликим, але вони відкривають світ додаткових датчиків і входів. Для точного керування в проекті можна використовувати потенціометри та аналогові джойстики. Чипи, такі як датчик температури TMP36, можна використовувати для збору точних даних для проекту. Якщо вашому проекту Arduino потрібно більше контактів GPIO, тоді Arduino Mega має 70 контактів, а плати клонів можна купити відносно дешево. Вартість: Raspberry Pi проти Arduino

Найдешевшим Raspberry Pi є Zero W, який продається за 10 доларів і забезпечує повноцінний комп'ютер Linux, з Wi-Fi і Bluetooth і доступом до важливого GPIO (хоча вам доведеться припаяти контакти самостійно). Використання Pi Zero W як вбудованого пристрою є недорогим способом створення проекту IoT. Найдорожчим Raspberry Pi є Pi 4 8 ГБ, який продається за 75 доларів, але щоб отримати максимальну віддачу від цієї плати, вам потрібно буде придбати додаткові аксесуари та плати HAT.

Вартість Arduino може варіюватися від кількох доларів за плату-клон до майже 100 доларів за офіційну Portenta H7. ATtiny85, недорогий мікроконтролер лише з шістьма контактами GPIO, можна купити менш ніж за 2 долари, і він пропонує достатню потужність для проектів робототехніки. Клони Arduino UNO можна купити відносно дешево і забезпечують досить хорошу сумісність у порівнянні з офіційною платою.

Різниця в ціні між клоном та офіційною платою Arduino відображає підтримку, яку компанія надає своїм спільнотам. Плати клонів не підтримують безпосередньо спільноту, але вони дешеві і в основному добре працюють. Офіційні плати працюють надзвичайно добре, і, купуючи плату, ви підтримуєте спільноту та Arduino у створенні нових продуктів та допоміжних матеріалів.

Arduino є явним переможцем за найнижчу вартість плати. Але це супроводжується парою застережень. Вам знадобиться комп'ютер для програмування Arduino, а також компоненти. Швидше за все, у вас вже є комп'ютер, тож це не пов'язано безпосередньо з витратами, вартість компонентів може сильно варіюватися, залежно від ваших вимог. Raspberry Pi — це власний комп'ютер, але йому все одно потрібні компоненти та доповнення, щоб максимально використовувати його. Хоча плати Arduino мають вбудовану пам'ять, для кожного Raspberry Pi потрібна карта microSD (перегляньте наш список найкращих карт microSD Raspberry Pi, або ви можете завантажитися з USB).

Підтримка мови програмування: Raspberry Pi проти Arduino

Raspberry Pi — це повноцінний настільний комп'ютер Linux, який просто має доступ до GPIO завдяки SoC Broadcom. Підключення Raspberry Pi до монітора, клавіатури та миші надає нам користувацький досвід, не дуже віддалений від звичайного комп'ютера.

Оскільки Raspberry Pi працює під керуванням Linux, він має доступ до багатьох різних мов програмування, деякі з яких також можна використовувати з GPIO. Python і Scratch — два очевидні приклади мов, які можуть працювати з GPIO, але є багато інших, включаючи Node-RED, Ruby і C.

З Arduino наш вибір дещо обмеженіший. Arduino IDE — це просто набір функцій C/C++, які компілюються та флешуються на плату. Плати Arduino призначені для підключення та програмування іншим комп'ютером, на якому працює IDE. Існують альтернативи Arduino IDE та мові C/C++. За допомогою перепрошивки спеціального ескізу (файлу проекту) в Arduino його можна також використовувати з Python або з мовами на основі блоків, такі як ArduBlockly і mBlock. Але це не дуже часто, і, враховуючи, що у вас є лише 16 КБ пам'яті для вашої програми на Arduino Uno, програми не можуть бути такі складні.

Заснований на безлічі варіантів, Raspberry Pi — це платформа з мовою для всіх можливих проектів. Існує багато мов, що охоплюють рівні кваліфікації та парадигми для проектів розробників, що включають GPIO, та мови для розробки програмного забезпечення, системного адміністрування та веб-розробки. Основні мови, особливо Python, є дуже популярними мовами, які мають масу бібліотек розширення та велику підтримку.

У Arduino є певний вибір, коли справа доходить до мов програмування, але це не повноцінний комп'ютер, і це обмежує кількість і тип коду, який ви можете кинути на нього.

Функціональність: Raspberry Pi проти Arduino

Raspberry Pi — це повнофункціональний настільний комп'ютер Linux, який можна використовувати для повсякденної діяльності або як сервер, але

він також забезпечує GPIO, який дозволяє використовувати комп'ютер у великих і малих проектах. Від простого готіння світлодіода до комп'ютерного зору, машинного навчання та робототехніки Raspberry Pi містить багато функцій у платі розміром з кредитну картку.

Насправді Raspberry Pi може робити все, що може зробити Arduino, але йому потрібна невелика допомога у вигляді NAT і додавання плат, оскільки деякі функції, такі як аналого-цифрове перетворення, не вбудовані.

Це правда, що Raspberry Pi не має такої швидкості, як Arduino. Зрештою, Raspberry Pi — це комп'ютер, якому необхідно завантажити операційну систему, перш ніж можна буде виконати будь-яку роботу, і, коли ви хочете його вимкнути, вам дійсно слід дати команду вимкнення та терпляче чекати, поки система вимкнеться.

Arduino — це єдина плата завдань, яка запускає одну програму за раз і відразу запускається, як тільки ви її ввімкнете, а коли ви захочете її вимкнути, ви можете просто витягнути вилку. Безпосередність Arduino є великою перевагою для проектів, які збирають дані або прості роботизації.

Arduino - це дійсно універсальна плата, але Raspberry Pi - це повноцінний комп'ютер і апаратна платформа для злому. Якщо вам потрібен бездротовий зв'язок, необроблена потужність обробки та доступ до GPIO, Raspberry Pi надає все це в невеликому пакеті.

Найкраще для навчання: Raspberry Pi проти Arduino

Хоча професіонали використовують їх для дуже серйозних кінцевих продуктів, Raspberry Pi і Arduino також розроблені для навчальних цілей, і це зрозуміло завдяки тисячам проектів і ресурсів, доступних в Інтернеті. Raspberry Pi є безсумнівним фаворитом для освіти, оскільки він працює з багатьма різними мовами програмування, може використовуватися для навчання основних обчислювальних концепцій і використовуватися як інструмент дослідження. І ви можете буквально підключити його та запустити без додаткового комп'ютера.

Екосистема Arduino орієнтована на написання коду на бажаній мові програмування для спілкування з платою. Він робить одну справу за раз, але робить це дуже добре. У середовищі класу Raspberry Pi буде домінувати завдяки своїй універсальності.

Крім того, те, що ви дізнаєтеся, працюючи з Raspberry Pi, нескінченно краще можна перенести на інші платформи. Python, найпопулярніша мова на Pi, працює на веб-серверах Windows, Mac і навіть Linux. Якщо ви не використовуєте плату Arduino або щось сумісне, знання коду Arduino не так корисно, як знання Python.

Можливо, у цьому перегоні було два коні, але один був набагато далі. Raspberry Pi відповідає потребам освіти. Він економний, простий у використанні, простий у зберіганні та може бути використаний для багатьох тем і проектів. Arduino — чудова плата для навчання, але в класі універсальність і простота використання є ключовою перевагою.

Висновки до розділу 2

У розділі 2 проаналізовано складові необхідні для апаратно-програмного комплексу з інтерактивною рекламою. Досліджено всі деталі Arduino Uno та світлодіодну матрицю.

Було проаналізовано всі відмінності між arduino та raspberry pi. Також показано всі сильні та слабкі сторони цих платформ.

Повністю досліджено світлодіодну матрицю та всі її можливості.

Було описано особливості плати Arduino її розпінівку. Були розглянуті різні контакти з нею та її програмування.

Також були зображені детальні схеми апарату.

РОЗДІЛ 3

ПРОГРАМНА ЧАСТИНА КОМПЛЕКСУ

В якості приладу для рекла обрано апаратно-програмний комплекс з пробігаючим рекламним оголошенням. Даний комплекс буде корисним через свою інтерактивність, адже можливо написати будь-який рекламний текст за декілька хвилин. Для того, щоб апаратний комплекс був конкурентноспроможний необхідно в майбутньому його вдосконалити й шукати шляхи розвитку.

3.1 Опис технології та мови програмування

Через те що у даному апаратному комплексі використовується плата Arduino , для його розробки обрано онлайн середовище розробки Wokwi(рис. 3.1).

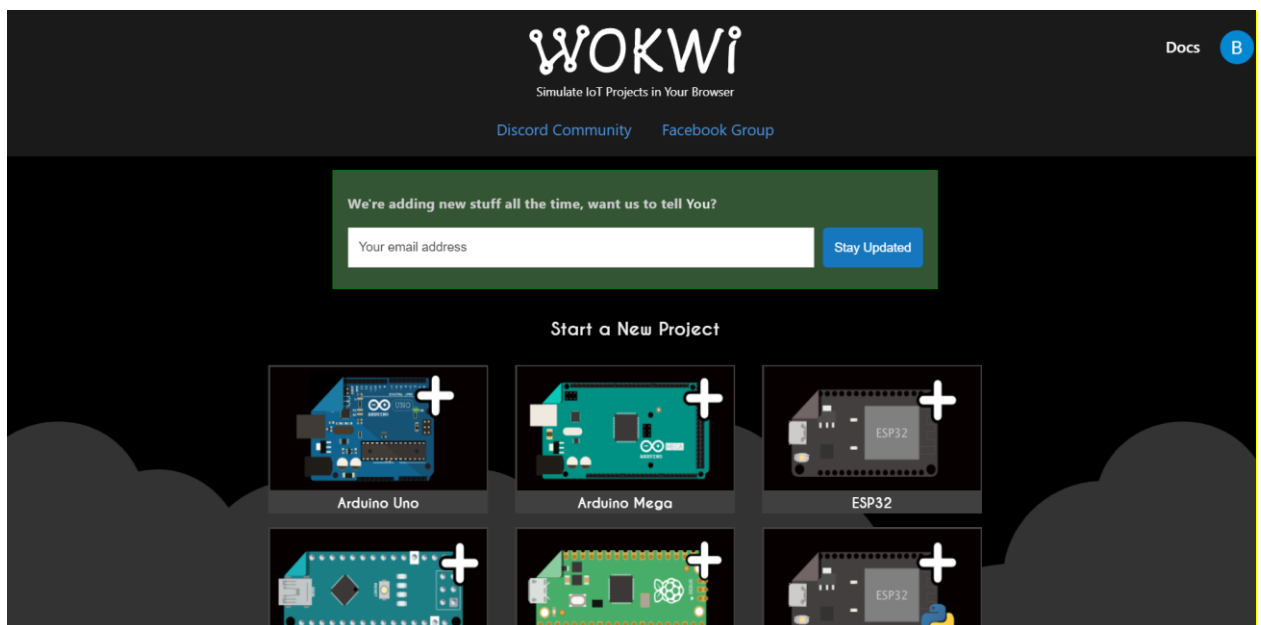


Рисунок 3.1 –Wokwi

Через простоту програм, які написані за допомогою Wokwi, їх називаються скетча . По суті, це текстові файли, написані мовою Arduino.

Arduino — це проектна, апаратна та програмна платформа з відкритим кодом, яка використовується для проектування та створення електронних пристроїв. Він розробляє та виробляє комплекти мікроконтролерів та одноплатні інтерфейси для створення електронних проектів.

Плати Arduino спочатку були створені, щоб допомогти студентам з нетехнічними знаннями.

У конструкції плат Arduino використовуються різноманітні контролери та мікропроцесори.

Плата Arduino складається з наборів аналогових і цифрових контактів вводу/виводу (введення/виводу), які додатково підключаються до макетної плати, плат розширення та інших схем . Такі плати мають модель універсальної послідовної шини (USB) та інтерфейси послідовного зв'язку , які використовуються для завантаження програм з комп'ютерів.

Також забезпечує проект IDE (Інтегроване середовище розробки), який базується на мові обробки для завантаження коду на фізичну плату.

Проекти авторизовані відповідно до GPL та LGPL . GPL називається загальнодоступною ліцензією GNU . Ліцензований LGPL називається GNU Lesser General Public License . Він дозволяє використовувати плати Arduino, це розповсюдження програмного забезпечення і може бути виготовлений будь-ким.

Також доступний у вигляді наборів для самостійної практики.

- Arduino використовується для різних цілей, таких як:
- Кнопка пальця
- Кнопка для активації двигуна
- Світло як датчики
- Світлодіодна кнопка
- Проектування
- Будівництво електронних пристроїв

Arduino - це програмна та апаратна платформа, яка допомагає створювати електронні проекти. Це платформа з відкритим вихідним кодом і

має різноманітні контролери та мікропроцесори. Існують різні типи плат Arduino, які використовуються для різних цілей.

Arduino - це єдина друкована плата, яка складається з різних інтерфейсів або частин. Плата складається з набору цифрових і аналогових контактів, які використовуються для підключення різних пристроїв і компонентів, які ми хочемо використовувати для функціонування електронних пристроїв.

Більша частина Arduino складається з 14 цифрових контактів вводу/виводу.

Аналогові контакти в Arduino в основному корисні для дрібнозернистого керування. Виводи на платі Arduino розташовані за певним шаблоном. Інші пристрої на платі Arduino - це порт USB, невеликі компоненти (регулятор напруги або генератор), мікроконтролер, роз'єм живлення тощо.

Нижче наведено характеристики Arduino:

–Програмування на Arduino — це спрощена версія C++, що полегшує процес навчання.

–Arduino IDE використовується для управління функціями плат. Далі він надсилає набір специфікацій мікроконтролеру.

–Arduino не потребує додаткової плати чи частини для завантаження нового коду.

–Arduino може зчитувати аналогові та цифрові вхідні сигнали.

–Апаратна та програмна платформа проста у використанні та реалізації.

3.1.1 Вимоги до синтаксису

Що стосується синтаксису, то він не схожий на синтаксис C++. Перша подібність, яку можливо помітити – це використання фігурних дужок для обгортання блоків коду. Якщо пропустити фігурну дужку, що закривається, після використання тієї, що відкривається, система видасть по лку. Arduino IDE виділить закриваючу дужку, якщо натиснути на відкриваючу, так що це

досить проста річ для перевірки. Як і C ++, Arduino також вимагає закінчувати твердження крапкою з комою. Пропущення призводить до спрацьовування по лки.

Ще одна чітка подібність - це спосіб введення коментарів. Існує два способи зробити це мовою Arduino, залежно від того, чи потрібен однорядковий чи блокований коментар. Якщо потрібно прокоментувати лише один рядок, необхідно розпочати його з двох похилих рисок:

```
// a comment here  
#define LED_PIN  
void setup() {  
  pinMode(LED_PIN, OUTPUT);
```

Якщо одного рядка замало для приміток, можливо вставити багаторядковий коментар, починаючи його косою рисою та зірочкою, закінчуючи зірочкою та косою рисою:

```
/* a comment here  
a comment there  
there are comments everywhere */  
#define LED_PIN  
void setup() {  
  pinMode(LED_PIN, OUTPUT);
```

Додаючи коментарі, необхідно пам'ятати, що компілятор Wokwi повністю їх ігноруватиме. Це означає, що він не експортуватиме їх у процесор і не використовуватиме пам'ять мікроконтролера [15].

3.1.2 Опис коду.

1. Вмикаємо оператори налагодження для послідовного виводу.

```
#if DEBUG
#define PRINT(s, x) { Serial.print(F(s)); Serial.print(x); }
#define PRINTS(x) Serial.print(F(x))
#define PRINTX(x) Serial.println(x, HEX)
#else
#define PRINT(s, x)
#define PRINTS(x)
#define PRINTX(x)
#endif
```

2. Визначаємо кількість пристроїв у ланцюжку та апаратний інтерфейс.

```
#define HARDWARE_TYPE MD_MAX72XX::PAROLA_HW
#define MAX_DEVICES 8
#define CLK_PIN 13
#define DATA_PIN 11
#define CS_PIN 10
```

3. Апаратне забезпечення SPI

```
MD_Parola P = MD_Parola(HARDWARE_TYPE, CS_PIN, MAX_DEVICES);
```

4. Параметри прокрутки.

```
#if USE_UI_CONTROL
const uint8_t SPEED_IN = A5;
const uint8_t DIRECTION_SET = 9; // change the effect
const uint8_t INVERT_SET = 8; // change the invert

const uint8_t SPEED_DEADBAND = 5;
#endif
```

5. Значення затримки кадру за замовчуванням
uint8_t scrollSpeed = 25;

6. Глобальні буфери повідомлень, які спільно використовують функції Serial і

Scrolling

```
#define BUF_SIZE 100
char curMessage[BUF_SIZE] = { "The best university in Ukraine - Chmnu. We are
waiting for you" };
char newMessage[BUF_SIZE] = { "waiting for you" };

bool newMessageAvailable = false;

#if USE_UI_CONTROL

MD_UISwitch_Digital uiDirection(DIRECTION_SET);
MD_UISwitch_Digital uiInvert(INVERT_SET);
```

7. Встановлюємо швидкість, якщо вона змінилася

```
{
    int16_t speed = map(analogRead(SPEED_IN), 0, 1023, 10, 150);

    if ((speed >= ((int16_t)P.getSpeed() + SPEED_DEADBAND)) ||
        (speed <= ((int16_t)P.getSpeed() - SPEED_DEADBAND)))
    {
        P.setSpeed(speed);
        scrollSpeed = speed;
        PRINT("\nChanged speed to ", P.getSpeed());
    }
}
```

8. Напрямок прокрутки та його зміна

```
if (uiDirection.read() == MD_UISwitch::KEY_PRESS)
{
    PRINTS("\nChanging scroll direction");
    scrollEffect = (scrollEffect == PA_SCROLL_LEFT ? PA_SCROLL_RIGHT :
PA_SCROLL_LEFT);
    P.setTextEffect(scrollEffect, scrollEffect);
    P.displayClear();
    P.displayReset();
}
```

9. Режим INVERT

```
if (uiInvert.read() == MD_UISwitch::KEY_PRESS)
{
    PRINTS("\nChanging invert mode");
    P.setInvert(!P.getInvert());
}
}
#endif
```

10. Цикл переміщення

```
while (Serial.available())
{
    *cp = (char)Serial.read();
    if ((*cp == '\n') || (cp - newMessage >= BUF_SIZE-2)) // end of message
character or full buffer
    {
        *cp = '\0'; // end the string
        // restart the index for next filling spree and flag we have a message
waiting
        cp = newMessage;
        newMessageAvailable = true;
    }
    else // move char pointer to next position
        cp++;
}
```

11. Місце введення рекламного оголошення

```
char curMessage[BUF_SIZE] = { "The best university in Ukraine - Chmnu. We are
waiting for you" };
char newMessage[BUF_SIZE] = { "waiting for you" };
```

3.2 Вибір компонентів АПЗ

Як і більшість інших мов кодування, мова Arduino дозволяє імпортувати зовнішні бібліотеки. Якщо вбудованих бібліотек недостатньо, можливо завантажити їх в Інтернеті або навіть написати власні.

Якщо коротко, бібліотека - це набір заздалегідь написаного коду, який надає вам додаткові функції.

Структура бібліотеки:

- бібліотека - це папка, що складається з файлів з файла коду C ++ (.cpp) і файлів заголовків C ++ (.h);
- файл .h описує структуру бібліотеки і оголошує все її змінні і функції;
- файл .cpp містить реалізацію функції.

Можливо використовувати як бібліотеки C, так і ті, які специфічні для Arduino. Вибравши бібліотеку, необхідно встановити її. Щоб включити певну бібліотеку у свій скетч, необхідно використати оператор #include і викликати бібліотеку, яку потрібно використовувати. Важливо пам'ятати, що не слід додавати крапку з комою: це твердження не потрібно припиняти.

Бібліотеки використані в апаратному комплексі

```
#include <MD_Parola.h>
#include <MD_MAX72xx.h>
#include <SPI.h>
#include <MD_UISwitch.h>|
```

3.2.1 Бібліотека MD_Parola.h

Parola — це модульний текстовий дисплей з прокручуванням, що використовує світлодіодні матричні контролери MAX7219 або MAX7221 на Arduino. Дисплей складається з будь-якої кількості однакових модулів, які підключаються разом, щоб створити ширший/довший дисплей.

- Вирівнювання тексту ліворуч, праворуч або по центру на дисплеї
- Прокрутка тексту, ефекти входу та виходу
- Керуйте параметра відображення та швидкістю анімації
- Підтримка апаратного інтерфейсу SPI
- Кілька віртуальних дисплеїв (зон) у кожному рядку світлодіодних модулів
- Визначені користувачем шрифти та/або окремі заміни символів
- Підтримка дисплеїв подвійної висоти
- Підтримка змішування тексту та графіки на одному дисплеї

Метою було створити світлодіодний матричний дисплей, схожий на Lego, із використанням стандартних світлодіодних матриць 8x8. Програмне забезпечення підтримує цю гнучкість завдяки масштабованому підходу, який вимагає лише визначення кількості модулів для адаптації існуючого програмного забезпечення до нової конфігурації.

3.2.1 Бібліотека MAX72xx

Бібліотека реалізує функції, які дозволяють використовувати MAX72xx для світлодіодних матриць (64 окремих світлодіодів), що дозволяє програмісту використовувати світлодіодну матрицю як піксельний пристрій, відображаючи графічні елементи так само, як і будь-який інший піксельний адресний дисплей.

У цьому сценарії зручно абстрагувати концепцію апаратного пристрою та створити однорідний і послідовний адресний простір пікселів з бібліотекою, які визначають адресу пристрою та елемента пристрою. Аналогічно, управління пристроєм є однорідним і абстрагованим до системного рівня.

Бібліотека все ще зберігає гнучкість для керування рівнем пристрою, якщо цього вимагає розробник, завдяки використанню методів перевантаженого класу.

3.2.2 Бібліотека SPI

Бібліотека SPI дозволяє спілкуватися з одним або кількома пристроями SPI (Serial Peripheral Interface). Підтримується тільки головний режим SPI для керування периферійною мікросхемою SPI.

Часто SPI використовується іншою бібліотекою (наприклад, Ethernet), які забезпечують легкий доступ до певного пристрою SPI. Хоча ви можете використовувати SPI безпосередньо, інші бібліотеки, які додають особливості чіпа, частіше використовуються.

3.2.3 Бібліотека MD_UISwitch.h

Ця бібліотека однорідно інкапсулює використання різних типів перемикачів для пристроїв введення користувача. Бібліотека легко розширюється для

додаткових типів комутаторів за допомогою ієрархії класів і моделі успадкування, дотримуючись коду для існуючих типів комутаторів.

Бібліотека включає в себе такі функції:

Програмне забезпечення для всіх типів комутаторів.

Автоматичне визначення натискання пере кача, подвійного натискання, тривалого натискання та автоматичного повтору.

Може працювати з перехода низький/високий або високий/низький.

Усі таймери можна програмно налаштувати - час відскоку, час подвійного натискання, час довгого натискання та час автоматичного повторення.

Бібліотека обробляє пере качі:

Моментні пере качі типу (клас MD_Switch_Digital)

Сигнали, керовані користувачем, наприклад, розширювачі вводу-виводу (клас MD_Switch_User)

Аналогові резисторні сходові пере качі (клас MD_Switch_Analog)

Матриця клавіатури (клас MD_Switch_Matrix)

Матриця клавіатури з використанням 4017 IC (клас MD_Matrix_4017KM)

3.3 Опис інтерфейсів АПЗ

3.3.1 Схеми підключення датчиків до Arduino

Спочатку обираємо Arduino Uno та натискаємо на нього.

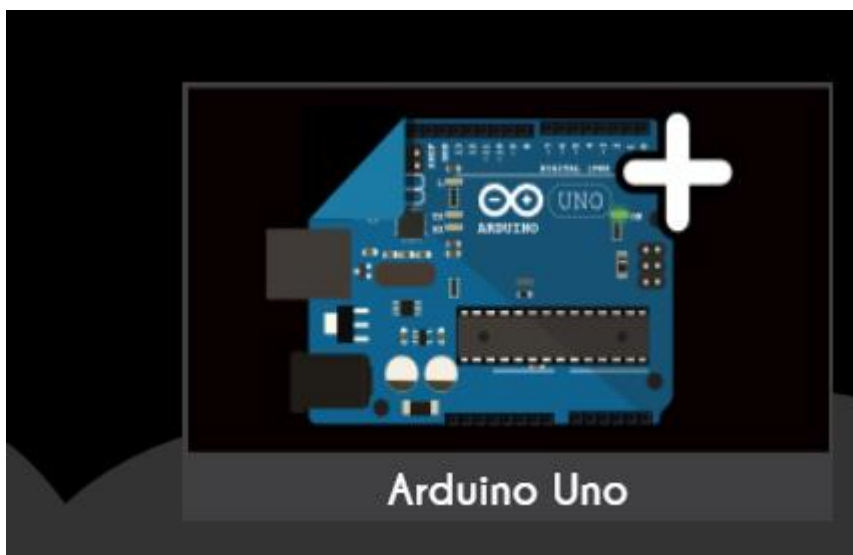


Рисунок 3.2 – Вигляд Arduino Uno

Потім обираємо світлодіодну матрицю та з'єднуємо їх

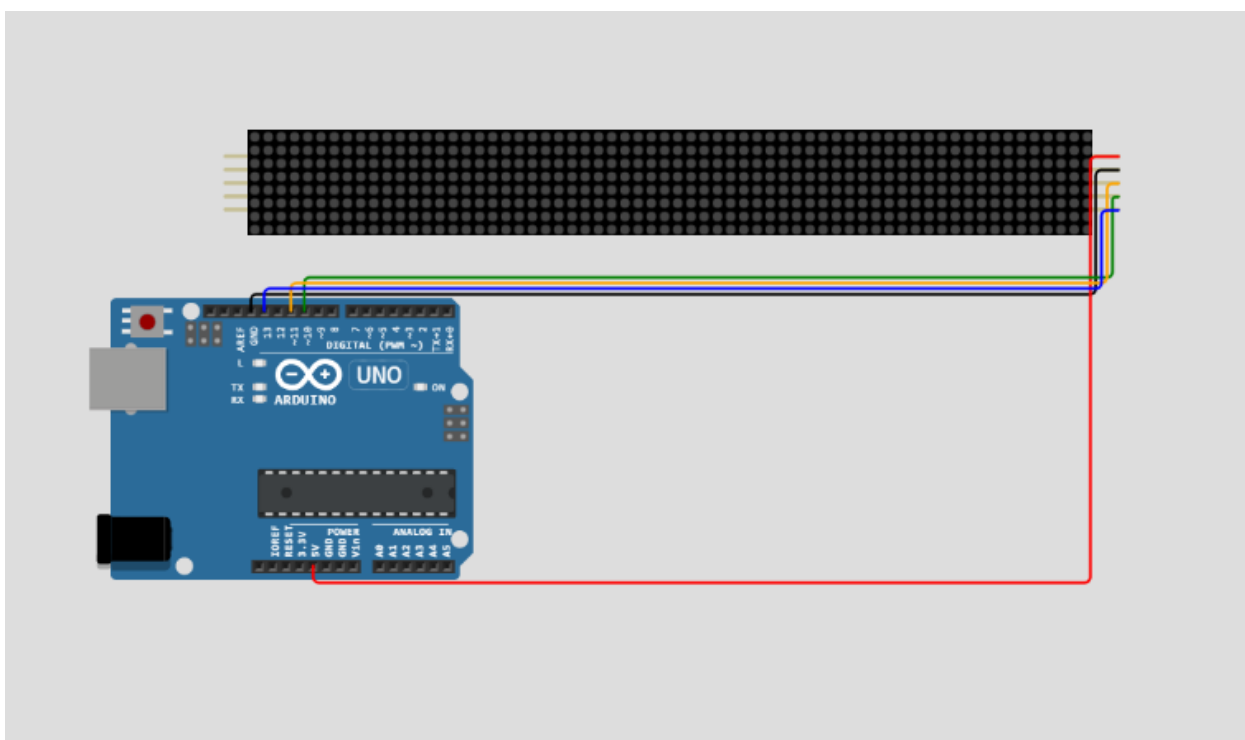


Рисунок 3.3 – Готовий програмний комплекс

3.4 Тестування приладу

Розроблений комплекс передбачає розгортання в окремому приміщенні для моніторингу, яка дозволить стежити за ефективністю рекламної компанії. Після аналізу отриманих даних споживачам установ де буде робитися

певний набір правил, що дасть змогу збільшити кількість споживачів та залучених людей .

Головний результат впровадження розробленого апаратно-програмного комплексу – підвищення якості інтерактивної реклами, зниження її вартості.

3.4.1 Експериментальне підтвердження

За допомогою розробленого приладу були проведені експеримент на вулиці, торговому центрі, в автобусі, в читальній залі і в спортивному залі, а також ресторані. Результати дослідження занесені в таблицю 3.1.

Таблиця 3.1 – Результати вимірювань

Місця розміщення	Кількість зацікавлених людей	Кількість людей, які скористались	%людей які скористались
Вулиця	7	4	57,14
Торговий центр	14	9	64,3
Автобус	9	6	66,7
Читальна зала	6	3	50
Спортивна зала	3	1	33,33
Ресторан	5	2	40
Магазин	11	7	63,64

3.4.2 Робота приладу

Написане ПЗ дозволяє апаратному комплексу працювати .Коли вводиться рекламне оголошення в потрібний для цього рядок.

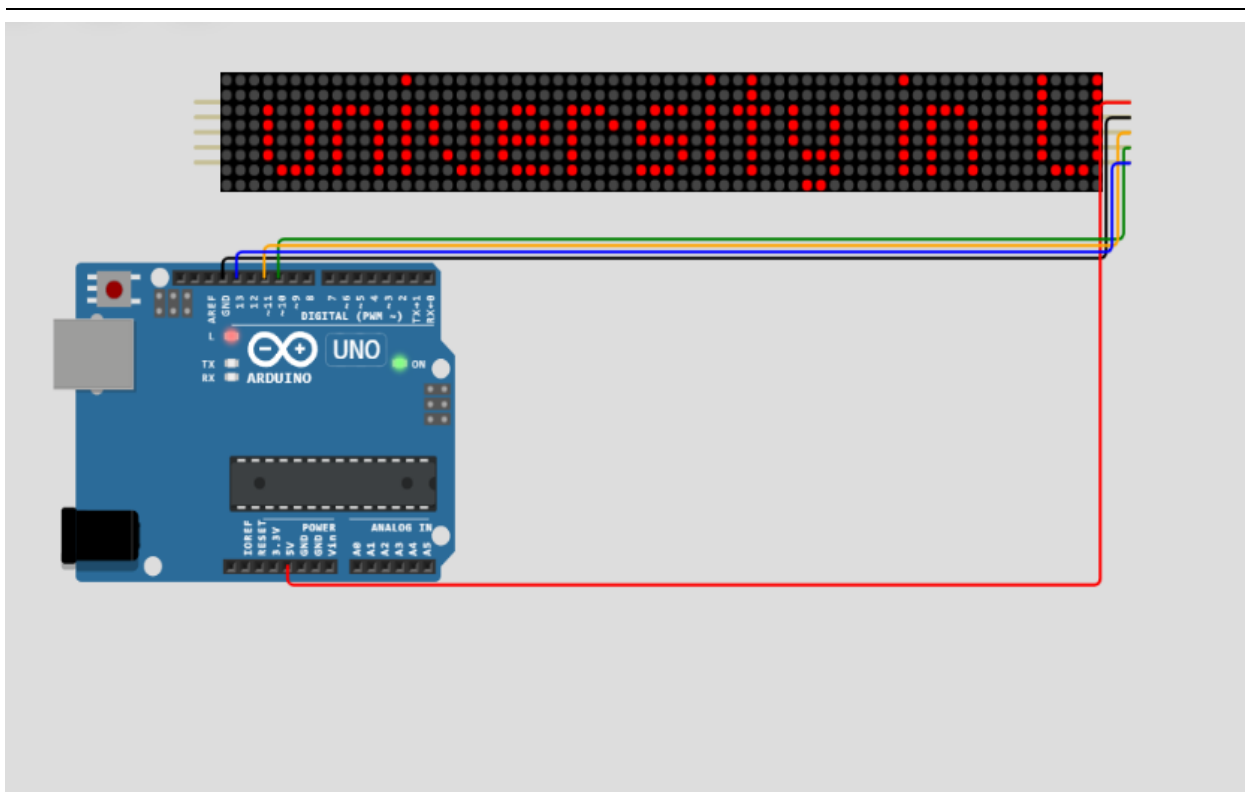


Рисунок 3.4– Прокручування комплексом рекламного оголошення

А в другому рядку потрібно ввести слова, які повинні донести всю суть рекламного оголошення . Й апарат зупиниться на цих словах, щоб зацікавити споживачів рекламного оголошення.

Саме ця частина апарату є найважливішою , адже реклама створена для того,щоб зацікавлювати людей .

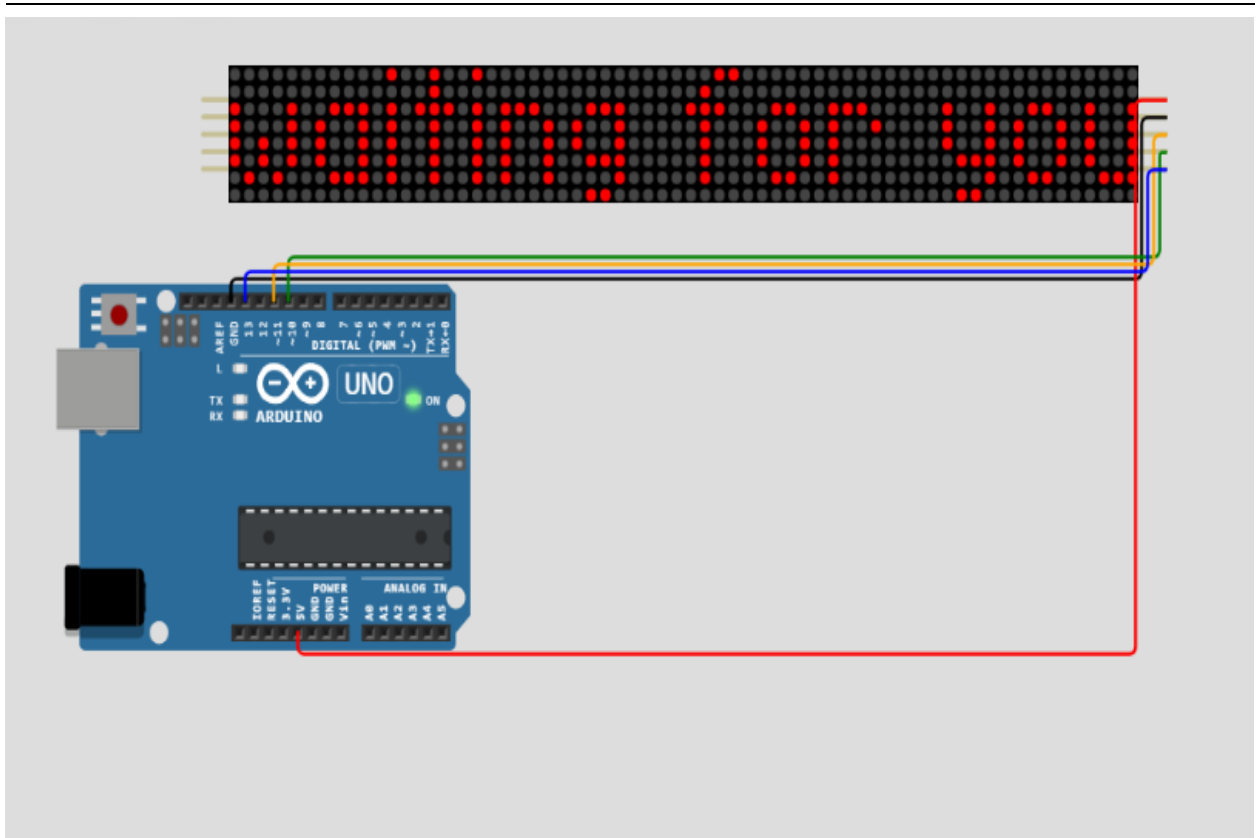


Рисунок 3.5– Зупинка на ключових словах

Висновки до розділу 3

В цій частині роботи була обрана мова програмування та компоненти до апаратного комплексу. Через те що апаратний комплекс виконано онлайн, то замість звичної Arduino IDE обрано онлайн середовище розробки WOKWI.

Модель оперує інформацією заданою в комплексі. Показано результати збору комплексу, загальний вигляд.

Здійснені експерименти перевірки апаратного комплексу, подальше тестування виявило, що комплекс привертає увагу людей й дає певні результати. Таким чином, отримані результати роботи комплексу показують, що в ньому є певний потенціал й подальша розробка певних допрацювань є можливим.

Продемонстровано вигляд комплексу у стані пробігаючого рядка і стану показу ключових словосполучень.

ВИСНОВКИ

Розроблено програмно-апаратний комплекс для інтерактивної зовнішньої реклами на базі мікроконтролерної плати Arduino згідно поставленої мети та цілей.

Проаналізувавши різні види джерел, було визначено проблеми сучасних видів реклам, а саме: високу вартість, подекуди відсутню мобільність та неактуальність. Порівняння переваг існуючих видів виявило, що вартість, актуальність та мобільність – найважливіші критерії. Але якщо необхідна креативна, інтерактивна та максимально діюча реклама, ціна на неї дуже зростає це – головний недолік існуючих на ринку систем.

Підбір потрібних для комплексу компонентів був згідно їх характеристик та переваг над аналогами.

Був обраний мікроконтролер Arduino Uno та світлодіодна матриця MAX7219, оскільки це найкращий підбір комплектуючих для створення комплексу, відповідного очікуванням від нього.

Для розроблення програмної частини обрано технологію Arduino та онлайн середовище розробки WOKWI, мову програмування Arduino, оскільки у даному проекті використовується плата Arduino Uno.

Тестування приладу виявило, що комплекс привертає увагу людей й дає певні результати. Таким чином, отримані результати роботи комплексу показують, що в ньому є певний потенціал й подальша розробка певних допрацювань є можливим. Модель оперує інформацією заданою в комплексі.

Головний результат впровадження розробленого апаратно-програмного комплексу – підвищення якості рекла , зниження вартості розміщення рекла , впровадження креативних ідей для оголошень.

Щодо певних удосконалень. Можливо додати можливість введення реклами через телефон .Зокрема можливе створення додатку й розміщення

його на App Store чи Google Play .Можливо також додати історію проведених рекламних компаній та результати по роботі з споживачами.

Через те що покращення чи удосконалення комплексу потребують додаткових витрат то було обрано найкраще співвідношення ціни до якості а саме: креативність, актуальність, інтерактивність і невелику вартість.Для додавання певних покращень та вдосконалень потрібне бажання замовника реклами та за їх необхідністю.

Було вирішене питання охорони праці та безпеки життєдіяльності після детального ознайомлення та вивчення їх.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. What is an Arduino?. URL: <https://learn.sparkfun.com/tutorials/what-is-an-arduino/all>(date of access: 02.03.2022).
2. What is JSON? URL: <https://developers.squarespace.com/what-is-json> (date of access: 10.05.22)
3. Arduino i Raspberry Pi. URL: <http://edurobots.ru/2014/09/arduino-ili-raspberry-pi-kakaya-platforma-luchshe/> (date of access:24.05.19).
4. Arduino cc. URL: <https://www.arduino.cc/> (date of access: 03.06.2022).
5. Робота з JSON та його стандартний вид. URL: <https://developer.mozilla.org/ru/docs/Learn/JavaScript/Objects/JSON> (date of access 01.04.2022).
6. H. Kimura, Y. Fukuoka, A.H. Cohen Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts Int. J. Robot Res., 26 (5) (2007), pp. 475-490
7. P.T. Doan, H.D. Vo, H.K. Kim, S.B. Kim A new approach for development of quadruped robot based on biological concepts Int. J. Precis. Eng. Manuf., 11 (4) (August 2010), pp. 559-568
8. Hardware and software specifications. Unreal Engine 5 Documentation | Unreal Engine 5.0 Documentation. URL: <https://docs.unrealengine.com/4.27/en-US/Basics/InstallingUnrealEngine/RecommendedSpecifications/> (date of access: 01.06.2022).
9. Е. Кучера. "Зв'язок між єдністю 3D-двигуна та мікроконтролером arduino: віртуальний розумний будинок" / Е. Кучера, О. Гаффнер та Ш. Козак // Кібернетика та інформатика (К&І) - 2018
10. Mini Thermocouple (Thermal Resistance). PT100 Datasheet. URL: <http://www.wr-wz.com/common/down/name/580d5472a1a32.pdf.html> (date of access: 03.06.2022).

-
11. RC1602B-datasheet. URL: https://www.raystar-optronics.com/upload_files/monochrome-lcd-module/16x2-character-lcd-display/RC1602B-datasheet.pdf (date of access: 02.06.2022).
 12. Analog Pulse Width Modulation. Texas Instruments. URL: <http://www.ti.com/lit/ug/slau508/slau508.pdf> (date of access: 01.06.2022).
 13. Среда разработки Arduino | Аппаратная платформа Arduino: URL: http://arduino.ru/Arduino_environment (date of access: 06.06.2022).
 14. Allan A. «Distributed Network Data» / A. Allan, K. Bradford / O'Reilly Media, Inc., 2013. — 168 pages.
 15. Anderson R. Pro Arduino (+source code) / R. Anderson., D. Cervo / Apress, 2013. - 305 p.
 16. Application Note: Event-Driven Arduino. Programming with QP: Document Revision H Quantum Leaps, LLC, July 2013. - 34 p.
 17. Böhmer M. Beginning Android ADK with Arduino / Böhmer M. / Apress. 2012.- 310 c.
 18. Evans B. Beginning Arduino Programming / B. Evans / - Apress, 2011. - 270 p.
 19. Karvinen T. Make a Mind-Controlled Arduino Robot / T. Karvinen, K. Karvinen / Brain as a Remote O'Reilly Media, 2011. - 96 p.
 20. Kelly J.F. Arduino Adventures: Escape from Gemini Station / J.F. Kelly, H. Timmis / Apress, 2013. - 332 p.

Додаток А

Лістинг програ

```
#include <MD_Parola.h>
#include <MD_MAX72xx.h>
#include <SPI.h>

// set to 1 if we are implementing the user interface pot, switch, etc
#define USE_UI_CONTROL 0

#if USE_UI_CONTROL
#include <MD_UISwitch.h>
#endif

// Turn on debug statements to the serial output
#define DEBUG 0

#if DEBUG
#define PRINT(s, x) { Serial.print(F(s)); Serial.print(x); }
#define PRINTS(x) Serial.print(F(x))
#define PRINTX(x) Serial.println(x, HEX)
#else
#define PRINT(s, x)
#define PRINTS(x)
#define PRINTX(x)
#endif

// Define the number of devices we have in the chain and the hardware
interface
// NOTE: These pin numbers will probably not work with your hardware and may
// need to be adapted
#define HARDWARE_TYPE MD_MAX72XX::PAROLA_HW
#define MAX_DEVICES 8
#define CLK_PIN 13
#define DATA_PIN 11
#define CS_PIN 10

// HARDWARE SPI
MD_Parola P = MD_Parola(HARDWARE_TYPE, CS_PIN, MAX_DEVICES);
// SOFTWARE SPI
//MD_Parola P = MD_Parola(HARDWARE_TYPE, DATA_PIN, CLK_PIN, CS_PIN,
MAX_DEVICES);

// Scrolling parameters
#if USE_UI_CONTROL
```

```
const uint8_t SPEED_IN = A5;
const uint8_t DIRECTION_SET = 9; // change the effect
const uint8_t INVERT_SET = 8;    // change the invert

const uint8_t SPEED_DEADBAND = 5;
#endif // USE_UI_CONTROL

uint8_t scrollSpeed = 25; // default frame delay value
textEffect_t scrollEffect = PA_SCROLL_LEFT;

textPosition_t scrollAlign = PA_LEFT;
uint16_t scrollPause = 1500; // in milliseconds

// Global message buffers shared by Serial and Scrolling functions
#define BUF_SIZE 100
char curMessage[BUF_SIZE] = { "The best university in Ukraine - Chmnu. We are
waiting for you" };
char newMessage[BUF_SIZE] = { "waiting for you" };

bool newMessageAvailable = false;

#if USE_UI_CONTROL

MD_UISwitch_Digital uiDirection(DIRECTION_SET);
MD_UISwitch_Digital uiInvert(INVERT_SET);

void doUI(void)
{
    // set the speed if it has changed
    {
        int16_t speed = map(analogRead(SPEED_IN), 0, 1023, 10, 150);

        if ((speed >= ((int16_t)P.getSpeed() + SPEED_DEADBAND)) ||
            (speed <= ((int16_t)P.getSpeed() - SPEED_DEADBAND)))
        {
            P.setSpeed(speed);
            scrollSpeed = speed;
            PRINT("\nChanged speed to ", P.getSpeed());
        }
    }
}

if (uiDirection.read() == MD_UISwitch::KEY_PRESS) // SCROLL DIRECTION
{
    PRINTS("\nChanging scroll direction");
    scrollEffect = (scrollEffect == PA_SCROLL_LEFT ? PA_SCROLL_RIGHT :
PA_SCROLL_LEFT);
    P.setTextEffect(scrollEffect, scrollEffect);
}
```



```
    P.displayClear();
    P.displayReset();
}

if (uiInvert.read() == MD_UISwitch::KEY_PRESS) // INVERT MODE
{
    PRINTS("\nChanging invert mode");
    P.setInvert(!P.getInvert());
}
}
#endif // USE_UI_CONTROL

void readSerial(void)
{
    static char *cp = newMessage;

    while (Serial.available())
    {
        *cp = (char)Serial.read();
        if ((*cp == '\n') || (cp - newMessage >= BUF_SIZE-2)) // end of message
            character or full buffer
            {
                *cp = '\0'; // end the string
                // restart the index for next filling spree and flag we have a message
                waiting
                cp = newMessage;
                newMessageAvailable = true;
            }
        else // move char pointer to next position
            cp++;
    }
}

void setup()
{
    Serial.begin(57600);
    Serial.print("\n[Parola Scrolling Display]\nType a message for the
scrolling display\nEnd message line with a newline");

#ifdef USE_UI_CONTROL
    uiDirection.begin();
    uiInvert.begin();
    pinMode(SPEED_IN, INPUT);

    doUI();
#endif // USE_UI_CONTROL

    P.begin();
    P.displayText(curMessage, scrollAlign, scrollSpeed, scrollPause,
scrollEffect, scrollEffect);
```

```
}

void loop()
{
#if USE_UI_CONTROL
  doUI();
#endif // USE_UI_CONTROL

  if (P.displayAnimate())
  {
    if (newMessageAvailable)
    {
      strcpy(curMessage, newMessage);
      newMessageAvailable = false;
    }
    P.displayReset();
  }
  readSerial();
}
```