

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Тернопільський національний економічний університет**  
**Факультет комп'ютерних інформаційних технологій**  
Кафедра комп'ютерної інженерії

Каштелян Іван Володимирович

**Пристрій послідовного виводу цифрової  
інформації на LCD-дисплей / Hardware of serial  
digital information output on the LCD display**

спеціальність: 6.050102 - Комп'ютерна інженерія  
освітньо-професійна програма - Комп'ютерні системи та мережі

Випускна кваліфікаційна робота

Виконав: студент групи КСМ-41/1  
Каштелян Іван Володимирович

Науковий керівник  
Масляк Б.О.

ТЕРНОПІЛЬ 2019

## РЕЗЮМЕ

Дипломний проект містить 64 сторінок пояснюючої записки, 31 рисунок, 7 таблиць, 3 додатки. Обсяг графічного матеріалу 3 аркуші А3.

Метою дипломної роботи є програмно-апаратна реалізація пристрою послідовного виведення інформації на LCD-дисплей.

Методи досліджень – методи побудови моделей та алгоритмів роботи цифрових пристроїв.

В дипломній роботі на основі аналізу програми курсу комп'ютерна схемотехніка та комп'ютерні системи навчальної та наукової літератури поставлена та реалізована задача програмно - апаратної реалізації пристрою послідовного виведення інформації на LCD дисплей. В роботі послідовно розглянуті питання сучасних тенденцій в побудові пристроїв візуалізації та відзначено актуальність їх застосування. Проектування структури передбачало побудову системи, детальний аналіз роботи інтерфейсу I2C та розробку алгоритму виводу інформації на LCD дисплей. В середовищі Fritzing побудовано схему апаратної реалізації пристрою, а в Arduino IDE розроблено та відлагоджено програмний скетч.

Ключові слова: ARDUINO, ДИСПЛЕЙ, ІНТЕРФЕЙС, ПІДКЛЮЧЕННЯ, ВИВІД ІНФОРМАЦІЇ, МІКРОКОНТРОЛЕР, ПРОГРАМНИЙ КОД.

## RESUME

The diploma project contains 64 pages of the explanatory note, 31 figures, 7 tables, 3 annexes. The volume of graphic material is 3 sheets A3.

The purpose of the thesis is the software and hardware implementation of the device serial output information on the LCD display.

Methods of research - methods of constructing models and algorithms of digital devices.

In the dissertation work on the basis of the analysis of the program of the course computer circuitry and computer systems of educational and scientific literature, the task of the software - hardware implementation of the device of serial information output on the LCD display is realized and realized. In the work consistently considered the issues of modern trends in the construction of visualization devices and noted the relevance of their application. The design of the structure involved the construction of a system, a detailed analysis of the I2C interface and the development of an algorithm for output information on the LCD display. In the Fritzing environment, a hardware implementation scheme was built, and a software sketch was developed and arranged in the Arduino IDE.

Keywords: ARDUINO, DISPLAY, INTERFACE, CONNECTION, SELECTION OF INFORMATION, MICROCONTROLLER, SOFTWARE CODE.

## ЗМІСТ

Вступ.....	10
1. Застосування мікроконтролерів в системах автоматизації .....	11
1.1 Аналіз методів побудови мікроконтролерних систем.....	11
1.2 Особливості функціонування периферійних пристроїв.....	17
1.3 Постановка задачі по розробці пристрою виведення інформації на LCD дисплей.....	22
2. Проектування програмно-апаратної структури пристрою .....	24
2.1 Структурно-функціональна схема та інтерфейс системи .....	24
2.2 Засоби візуалізації та методика налаштування системи .....	26
2.3 Алгоритм виводу інформації .....	29
2.4 Вибір середовища автоматизованої розробки системи виведення інформації .....	30
3. Реалізація пристрою послідовного виведення інформації на LCD дисплей .....	35
3.1 Апаратна реалізація пристрою .....	35
3.2 Написання програмного коду роботи пристрою .....	36
3.3 Налаштування та тестування .....	39
4. Техніко-економічний розділ.....	44
4.1 Розрахунок витрат на розробку програмного забезпечення .....	43
4.2 Визначення експлуатаційних витрат.....	49
4.3 Розрахування споживання проектного рішення .....	51
4.4 Визначення показників економічної ефективності .....	52
Висновки.....	55
Список використаних джерел.....	56

					БР.КСМ. 07090/15.00.00.000 ПЗ					
Змн.	Арк.	№ докум.	Підпис	Дата	<b>ПОСЛІДОВНИЙ ВИВІД ЦИФРОВОЇ ІНФОРМАЦІЇ НА LCD ДИСПЛЕЙ</b>					
Розроб.		Шумський А.З.						Літ.	Арк.	Акрушів
Перевір.		Масляк Б.О						8	63	
Консульт.		Паздрій І.Р.						ТНЕУ. ФКІТ. КСМ-41/1		
Н. Контр.		Гураль І.В								
Затвердив.		Березький О.М								

Додаток А Лістинг коду програми .....	59
Додаток Б Світлокопія публікації .....	61
Додаток В Довідка про використання.....	64

					КР.КСМ. 07090/15.00.00.000 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВСТУП

Інтенсивний розвиток мікроелектроніки і збільшення степені інтеграції відкрили нові напрямки в розвитку комп'ютерної техніки – створення мікропроцесорів та мікроконтролерів. Поява такої техніки дала поштовх в області не тільки обчислювальної техніки, а й в області систем управління обладнанням. Створення високопродуктивних і надійних цифрових мікроконтролерних систем дозволило здійснювати їх інтеграцію з об'єктом управління (робоча машина, електронний прилад, різні механізми) [15]. Невеликі розміри, гнучкість в модернізації програмного забезпечення дозволяє інтегрувати їх в різну апаратуру, що значно розширює її функціональні можливості. На сьогоднішній день існує більше 200 різновидів мікроконтролерів які випускаються понад двадцятьма різними компаніями [10]. Найбільш поширеними є мікроконтролери фірми Atmel. Велике різноманіття моделей забезпечує високу продуктивність роботи відповідно до вимог розробника.

Мікроконтролерні платформи найпростіший спосіб управління мікропроцесором. Розрізняють їх по потужності, різноманіттю підключуваної периферії, розмірами, та ціною. Серед них виділяються три найпопулярніші платформи: Arduino, Raspberry PI, BeagleBone. Кожна з яких має свій спектр виконуваних задач та користується попитом на ринку. А щоб розширити їх функціонал використовують так звані шилди – це модулі які розширюють функціонал плати покращуючи чи додаючи те, що не передбачено виробником чи упущено для зменшення ціни. Для спрощення взаємодії розробника з платформою було розроблено велику кількість середовищ розробки, для Arduino – це Arduino IDE в якому на мові C++ з легкістю реалізується взаємодія з датчиками, обробка та аналіз інформації та проведення нескладних обчислень [14]. В цей же час більш потужні платформи Raspberry PI, BeagleBone працюють на базі операційних систем сімейства Linux на яке вже

					БР.КСМ. 07090/15.00.00.000 ПЗ	
Зм.	Арк.	№ докум.	Підпис	Дата		10

безпосередньо встановлюються необхідні середовища розробки [9]. Саме через це вони вважаються більш складними для початківців. Але завдяки цьому з ними можна взаємодіяти на таких мовах як: Java, Python, C, Ruby, JavaScript.

Використання мікропроцесорної бази потребує нового підходу до проектування, технологій виробництва і експлуатації електронних апаратів, а також спеціальні вимоги до підготовки спеціалістів які повинні добре орієнтуватися в питаннях логічного проектування, конструювання апаратура, а також в питаннях зв'язаних з програмуванням виконуваних даним апаратом функцій.

Оскільки типові мікроконтролерні системи не мають вбудованих дисплеїв розробники не завжди можуть перевірити правильність роботи пристрою тому важливу роль має візуалізація інформації, яка допомагає проаналізувати проміжні чи кінцеві результати роботи пристрою. Велике різноманіття давачів інформації та можливість мікропроцесорних платформ перетворити цю інформацію в зрозумілу для людей створює необхідність візуалізації цієї інформації такими способами: звуковий сигнал різної гучності який розповсюджує динамік, яскравості світлодіода чи безпосередньо виведення значення на дисплей. Залежно від потреб розробник має достатньо способів для взаємодії системи з користувачем.

Розробка пристрою візуалізації інформації допомагає зрозуміти роботу пристрою та показати результати його роботи користувачу, важливим фактором роботи такою пристрою є його портативність. Використання мікропроцесорних платформ дає змогу взаємодіяти з широким спектром пристроїв які з легкістю можна підключити та взаємодіяти з ними.

# 1 ЗАСТОСУВАННЯ МІКРОКОНТРОЛЕРІВ В СИСТЕМАХ АВТОМАТИЗАЦІЇ

## 1.1 Аналіз методів побудови мікроконтролерних систем

Мікроконтролери знайшли широке застосування в різних сферах науки та техніки завдяки низькій вартості, невеликих габаритах, високій надійності і значним набором функцій. Це в поєднанні з можливістю адаптації мікроконтролерів для роботи в конкретному середовищі визначило актуальність їх повсюдного використання. Типова структура мікроконтролера приведена на рисунку 1.1.

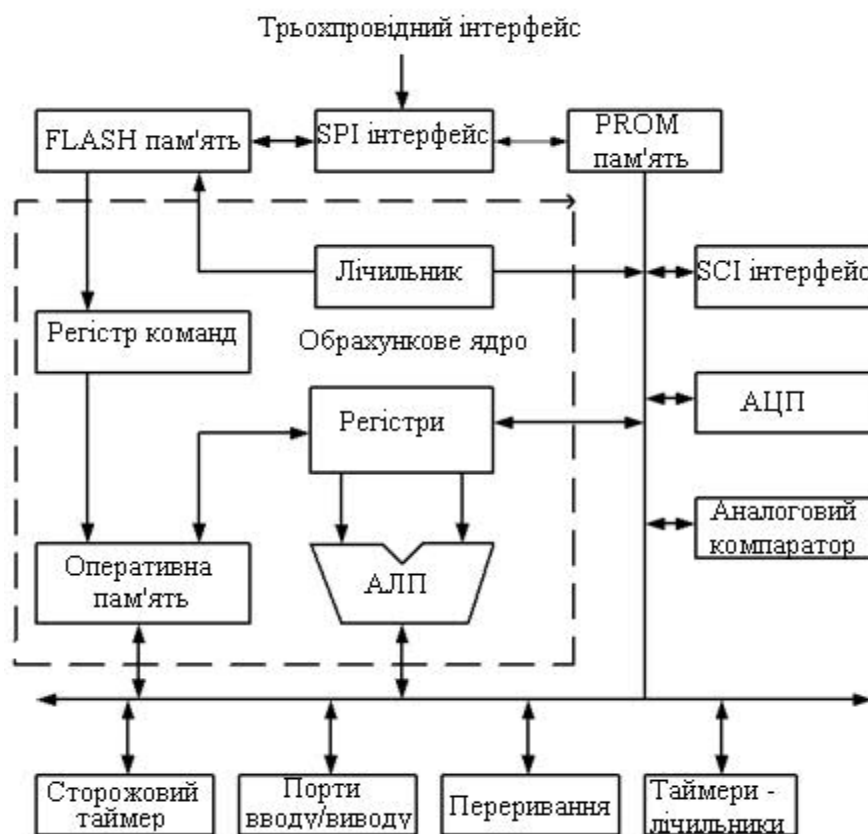


Рисунок 1.1 – Типова структура мікроконтролера

Мікроконтролер – це мікросхема яка представляє собою міні-комп'ютер, призначений для виконання різноманітних функцій. Дана мікросхема працює у відповідності до закладеної в неї програми, яку створює



програміст. Мікроконтролер може містити в собі велику кількість периферійних пристроїв які оприділяють його можливості, а також вартість [16]. Як правило будь-який мікроконтролер містить в собі наступні компоненти:

- арифметико-логічний пристрій;
- оперативна пам'ять;
- постійна пам'ять;
- генератор тактової частоти;
- порти вводу/вивода;
- таймери.

Лідерами у виробництві контролерів є такі гіганти як Atmel та Intel які постійно конкурують на ринку. Майже всі кристали промислового стандарту MSC-51 фірми Intel були замінені прямими аналогами сімейства AT89 фірми Atmel. На даний момент за підсумками останніх років велику частину ринку займають саме мікроконтролери Atmel. Та вже декілька років утримує перше місце з виробництва Flash-мікроконтролерів загального призначення. Усі мікроконтролери даної корпорації можна розділити на три великі групи рисунок 1.2.

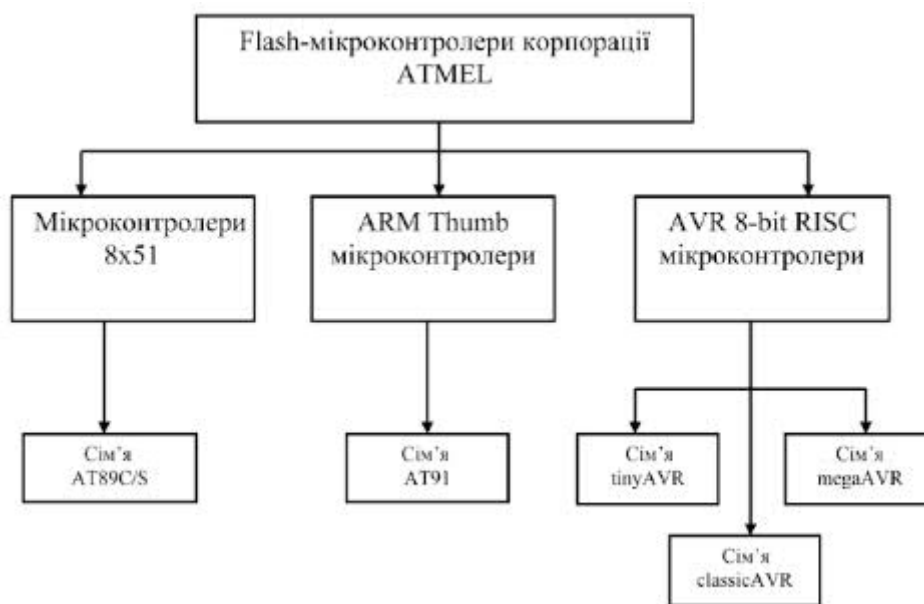


Рисунок 1.2 – Класифікація мікроконтролерів корпорації Atmel

Найсучасніша та найбільш популярна серія AVR 8-bit до теперішнього часу налічує понад 20 різних типів, які об'єднані в три основні групи:

- Mega AVR(префікс ATmegaXXX);
- Classic AVR(префікс AT90SXXX);
- Tiny AVR(префікс ATinyXXX).

Mega AVR мають найбільші об'єми пам'яті, найбільшу кількість виводів і найбільш повний набір периферійних вузлів. Можна назвати такі відмінності Mega AVR порівняно з Classic AVR – значні розміри вбудованої пам'яті програм і даних, практично повний набір периферійних вузлів, простий 8-розрядний таймер який має більш розвинуті режими роботи [13].

Одним з найпотужніших мікроконтролерів сімейства Mega AVR це ATmega 128. Це 8-розрядний мікроконтролер заснований на розширеній AVR архітектурі. За рахунок виконання більшості операцій за один машинний цикл досягає продуктивності 1млн. операцій в секунду, що дозволяє проектувальнику системи оптимізувати співвідношення енергоспоживання до швидкодії [13].

Однак для успішного функціонування мікроконтролера необхідний ряд додаткових стандартних пристроїв (контролери інтерфейсів, пристрої збору інформації), що створило передумови для розробки мікроконтролерних платформ. В даний час цей напрям представлений продукцією під назвою Arduino Uno, BeagleBone, RaspberryPi та великою кількістю їх аналогів.

Arduino Uno – Досить збалансована плата на якій розміщено все потрібне та нічого лишнього. На платі 14 цифрових пінів, які працюють на вхід та вихід. Підключається до комп'ютера за допомогою USB. Є роз'єм для окремого живлення, кнопка скиданн. Працює він на основі мікроконтролера ATmega328, має 14 цифрових входів/виходів, кварцевий резонатор тактовою частотою 16МГц, роз'єм USB, а також кнопку для скидання. Також на платі містяться відновлювані запобіжники, які захищають USB порт від коротких замикань та перегрузок. Якщо сила струму перебільшує 500мА тоді запобіжник автоматично розриває з'єднання з комп'ютером[13]. Великим

					БР.КСМ. 07090/15.00.00.000 ПЗ	14
Зм.	Арк.	№ докум.	Підпис	Дата		

плюсом цієї платформи є її ціна яка коливається від 5\$ до 10\$ яка є достатньо низькою в порівнянні з її конкурентами, саме невелика ціна та відносно великі можливості зробили цю мікроконтролерну систему такою популярною та потрібною на ринку (рисунок 1.3). Для неї доступні просте середовище розробки і велика база знань та готових проектів, що говорить про можливість створення достатньо функціональних додатків.

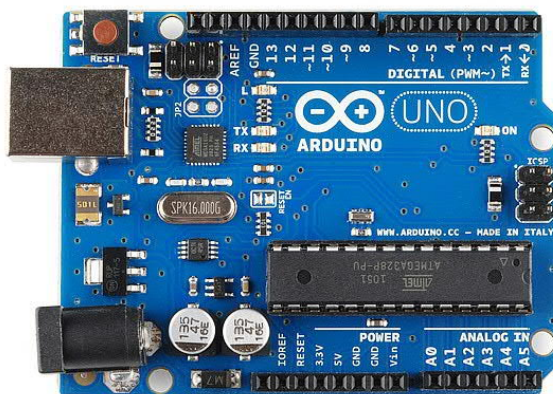


Рисунок 1.3 – Одноплатний комп'ютер Arduino Uno

Raspberry Pi перша модель з'явилася в 2012 році. Працює на операційній системі Linux. Дана мікроконтролерна система пропонує її власнику досить великий набір портів таких як usb, аудіо вихід, а у деяких моделях присутній hdmi порт.

Основна складність роботи з Raspberry Pi полягає в необхідності використання операційної системи найчастіше це Linux. Необхідність використовувати операційну систему несе за собою деякі складності, наприклад при неправильному запуску можуть виникнути деякі складності з роботою. Однак при виконанні цього чи іншого завдання є багато готових різноманітних програм. Також не потрібно забувати, що Raspberry Pi це дуже потужна система яка може виконувати багато завдань звичайного комп'ютера, але значно компактніша(рисунок 1.4).



Рисунок 1.4 – Одноплатний комп'ютер Raspberry Pi

BeagleBone є найменш відомою в порівнянні з попередньо наведеними платформами, але її можливості заслуговують уваги розробників програмного забезпечення та інтернет речей. Це потужний міні-комп'ютер який працює на Linux, компактних розмірів та підтримує роботу з Android і Ubuntu (рисунок 1.5).

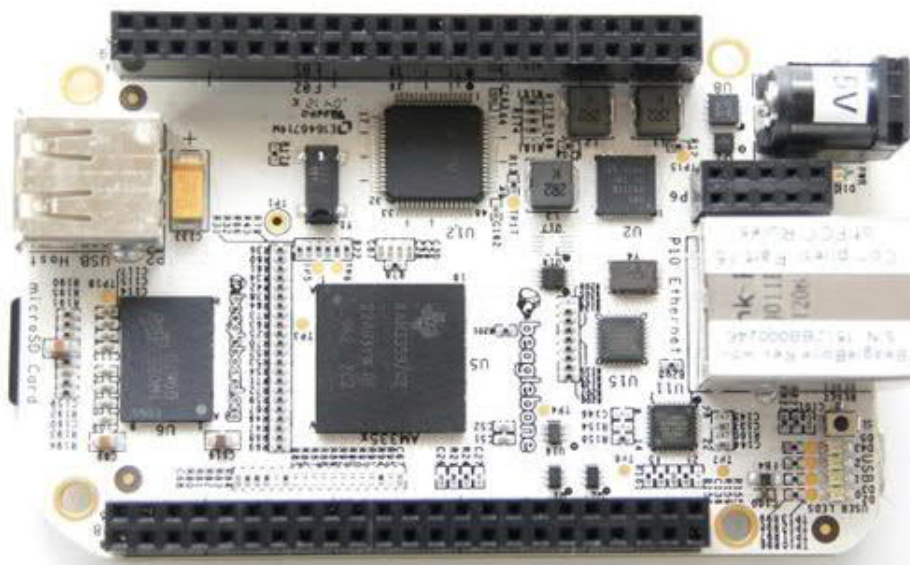


Рисунок 1.5 – Одноплатний комп'ютер BeagleBone

Вибираючи платформу для розробки слід звернути увагу на саме завдання та ретельно обдумати функціонал якої плати підходить більше для його роботи.

За допомогою Arduino та BeagleBone дуже легко взаємодіяти з різноманітними датчиками, двигунами та іншими електронними компонентами. Для керування необхідно тільки декілька рядків коду. Також основною перевагою платформи Arduino є невелике споживання енергії і легко підключення та налаштування зовнішніх модулів. Якщо проекту потрібна висока продуктивність то слід вибрати BeagleBone або Raspberry Pi.

Виходячи з аналізу ринку для реалізації проекту пропонується використати мікроконтролерну платформу Arduino Uno. Яка є простою в використанні, має невелику вартість та широкий функціонал, та достатньо продуктивності для поставленого завдання.

## 1.2 Особливості функціонування периферійних пристроїв

Виходячи з аналізу структури мікроконтролера розглянемо питання використання периферійних пристроїв. В загальному випадку вони поділяються на наступні групи:

- датчики;
- виконуючі пристрої (драйвери);
- засоби візуалізації.

Датчики використовують в системах автоматизації для збору інформації про стан об'єкту автоматизації. По призначенню датчики бувають для вимірювання:

- температури;
- тиску;
- вологості повітря;
- кислотності;
- освітлення;
- рівня води;

					БР.КСМ. 07090/15.00.00.000 ПЗ	17
Зм.	Арк.	№ докум.	Підпис	Дата		

– і багато інших спеціалізованих компонентів.

В свою чергу вказані вище можуть бути аналоговими, або ж цифровими, тобто при отриманні інформації мікроконтролер обробляє цифровий сигнал, або вимірює напругу з аналогового виходу [8]. Пізніше отриманий сигнал необхідно перетворити в форму, в якій вимірюється інформація і може бути сприйнята людиною. В автоматизованих системах керування датчики можуть виступати в ролі ініціюючих пристроїв вмикаючи обладнання чи приводячи в дію якийсь програмний алгоритм [14]. Типові датчики зображені на рисунку 1.6.



Рисунок 1.6 – Типові представники давачів інформації

Так звані модулі розширення – шилди, використовуються для розширення функціоналу та спрощення роботи розробника. Вони дають можливість за їх допомогою розробляти більш складні проекти [10]. Представляючи собою готові рішення для типової задачі. Модулі спрямовані лише на розширення функціоналу платформи, так як наприклад в Arduino uno немає встроєних Bluetooth, wi-fi чи Ethernet модулів, їх підключення доволі

просте обмежується тільки декількома проводами, а робота з ними максимально спрощена. Також широко використовуються різноманітні драйвери які підключаються окремо і являють собою мікросхему яка кріпиться на макетній платі та спрощує роботу з тим чи іншим модулем. Наприклад шилд для роботи з дисплеєм з декількома кнопками для керування зображень на рисунку 1.7.



Рисунок 1.7 – Шилдт дисплея з вбудованими кнопками

Мікроконтролери дозволяють зробити різні системи автоматизації та моніторингу. Але не потрібно забувати, що для взаємодії техніки та людини потрібні пристрої введення – різноманітні кнопки, потенціометри чи можливо інші пристрої, а також пристрої виведення – світлові індикатори( наприклад світлодіоди), звукові сигналізатори і звичайно дисплеї які в XXI столітті набули найбільшої популярності та розповсюдженості [10]. На даний момент існує декілька видів дисплеїв які можна розділити на такі види:

- сегментні (такі як на годинниках);
- графічні;
- алфавітно-цифрові.

Сегментні використовуються для індикації простих величин наприклад: температура, час, кількість обертів. Таків використовуються в калкуляторах та

на б'юджетній побутовій техніці і по сьогоднішній час. Інформація виводиться шляхом засвічування певних символів.

Графічні дисплеї ми зустрічаємо у повсякденному житті це наприклад монітор комп'ютера чи дисплей смартфона. На ці дисплеї виводиться будь яке зображення чи графічний інтерфейс з яким можна взаємодіяти користувачу. Головним елементом такого дисплею є пікселі [10].

Алфавітно-цифрові дисплеї ми зустрічаємо на вокзалах чи аеропортах, на них немає необхідності виводити зображення, а лише корисну інформацію про прибуття чи відправлення поїзда. Такі дисплеї виготовляються за допомогою технологій таких як LCD дисплеї тобто рідкокристалічні дисплеї принцип дії яких ґрунтується на явищі електронного переходу Фредерікса, а також LED дисплеї тобто світлодіодні в яких кожним пікселем виступає один або декілька напівпровідникових світлодіодів. Значною перевагою LED дисплеїв є велика яскравість, надійність оскільки пошкодження частити екарана не веде до його неприцездатності в цілому, до недоліків можна віднести велику вартість та так звану зернистість. Перевагою використання LCD дисплея є чітке і стабільне зображення, широкий вибір розмірів, великий термін служби, невелика вартість. Зображено на рисунку 1.8.

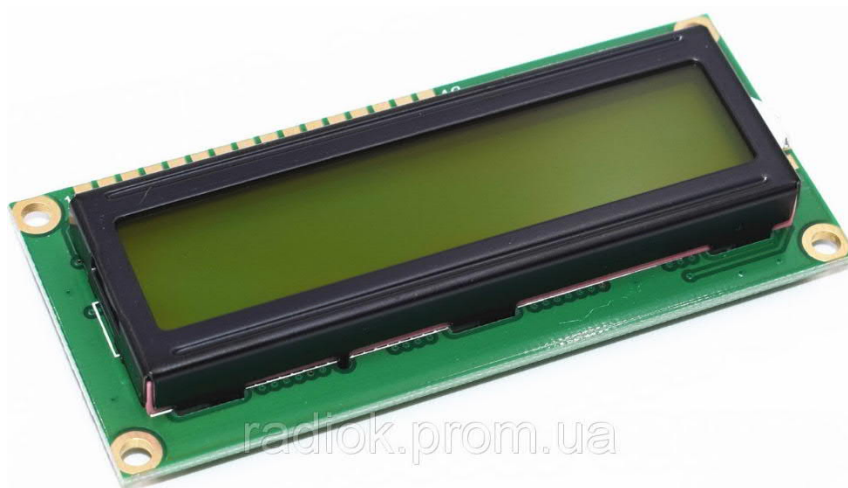


Рисунок 1.8 – LCD дисплей

					БР.КСМ. 07090/15.00.00.000 ПЗ	20
Зм.	Арк.	№ докум.	Підпис	Дата		



При підключенні можливі на вибір послідовний та паралельний інтерфейси. Як можна зрозуміти з назви при послідовному зв'язку дані передаються – послідовно, при паралельному – паралельно. Послідовна передача може бути організована з меншою кількістю проводів в порівнянні з паралельним методом комунікації, але буде потрібен певний механізм синхронізації для успішного зв'язку. Слід звернути увагу на те, що при послідовному лише передача даних відбувається послідовно, всі решту процеси такі як обробка даних відбуваються паралельно.

Послідовний зв'язок може класифікуватися на два види:

- синхронний;
- асинхронний.

При синхронному передатчик та приймач мають спільну систему тактів для синхронізації один з одним [21]. А синхронний тип не потребує спільного джерела синхронізації, дві сторони працюють у відповідності зі своїми незалежними системами тактів.

Для зв'язку використовуються наступні протоколи:

- SPI (Serial Peripheral Interface);
- I2C (Inter-Integrated Circuit).

SPI – протокол послідовного зв'язку синхронного типу, який складається з двох ліній даних, однієї тактової лінії, і лінії вибору підлеглих. В любий момент часу може бути лише один керуючий пристрій, підлеглі не можуть комунікувати один з одним, та не можуть відправляти дані без дозволу керуючого. У всьому процесі послідовне тактування грає велику роль оскільки кожний підлеглий пристрій залежить від цього годинника, щоб читати дані через одну лінію та відправляти через іншу [24].

Перевагами використання цього протоколу є можливість підключення та роботи з декількома пристроями та швидка форма послідовного зв'язку. Недоліками це те, що для кожного пристрою потрібна окрема лінія для кожного пристрою, тільки ведучий пристрій керує всіма процесами і підлеглі не можуть зв'язуватися один з одним.

					БР.КСМ. 07090/15.00.00.000 ПЗ	21
Зм.	Арк.	№ докум.	Підпис	Дата		

I2C – синхронний протокол зв'язку в якому використовуються тільки два кабелі для всього процесу, один для передачі послідовних даних, інший для передачі послідовного тактування. Може підтримувати декілька ведучих пристроїв на відміну від SPI. Кожен пристрій може відправляти чи отримувати дані використовуючи лише один кабель. Кожен підлеглий пристрій має свою власну унікальну адресу (від 7 до 10 біт) яку ведучий використовує для ідентифікації [5].

Перевагами використання цього протоколу: послідовний синхронний зв'язок який набагато надійніше асинхронного, може бути декілька ведучих пристроїв які можуть з'єднуватись один з одним, і використання для зв'язку тільки двох проводів, що робить його максимально компактним. Недоліком буде порівняно менша швидкодія аніж SPI [24].

В результаті розгляду пристроїв візуалізації для застосування пропонується використати LCD дисплей який буде давати чітке та стабільне зображення.

Таким чином, автономне застосування мікроконтролерних систем управління об'єктами автоматизації вимагає розробки та налаштування систем візуалізації.

### 1.3 Постановка задачі по розробці пристрою виведення інформації на LCD дисплей

Враховуючи результати аналізу параметрів, сфери застосування та вартості комплектуючих було виявлено, що в умовах навчального закладу найбільш оптимальним є використання мікроконтролера Arduino Uno який має достатню продуктивність та невелику вартість, алфавітно-цифрового дисплея виготовленого по технології LCD (рідкокристалічного) використовуючи послідовне синхронне підключення яке працює по протоколу I2C.

					БР.КСМ. 07090/15.00.00.000 ПЗ	22
Зм.	Арк.	№ докум.	Підпис	Дата		

Оскільки існує велика кількість схемотехнічних та програмних рішень для пристроїв виведення інформації на дисплей, було вибрано недорогий варіант реалізації.

Постановка завдання дипломного проекту повинна передбачати шлях для вирішення різних задач розробки. Тому другому розділі проекту поставлено наступні завдання:

- здійснити розробку структурно-функціональної системи візуалізації;
- виробити методику налаштування апаратної частини;
- розробити алгоритм виводу інформації;
- обґрунтувати вибір програмного засобу розробки програми;
- передбачити тестування програмної реалізації.

В третьому розділі проекту пропонується здійснити апаратну та програмну реалізацію пристрою послідовного виведення інформації на дисплей за допомогою послідовного інтерфейсу I2C та LCD дисплея 16x2.

При обґрунтуванні проектних рішень слід враховувати наступні вимоги до мікроконтролерної системи послідовного виводу інформації на LCD-дисплей:

- можливість оновлення прошивки;
- можливість багаторазового використання;
- простота управління;
- портативність.

Таким чином, в даному розділі сформовано вимоги та здійснена постановка задачі по розробці пристрою виводу інформації.

## 2 ПРОЕКТУВАННЯ ПРОГРАМНО-АПАРATНОЇ СТРУКТУРИ ПРИБРОЮ

### 2.1 Структурно-функціональна схема та інтерфейс системи

Для виконання даного проекту було використано одноплатний комп'ютер фірми Arduino, на базі мікроконтролера ATmega 328. Його технічні характеристики такі:

- флеш пам'ять 16кБ;
- ОЗУ 2кБ;
- постійна пам'ять даних 1 кБ;
- таймери 2x8 біт;
- лічильник 16 біт.

Мікроконтролер працює на основі гарвардської архітектури рисунок 2.1 тому має можливість одночасно працювати з пам'яттю програм та пам'яттю даних, що суттєво збільшує продуктивність.



Рисунок 2.1 – Принцип роботи гарвардської архітектури в мікроконтролері ATmega328

Завдяки гарвардській архітектурі є можливість одночасного зчитування коду інструкції та даних тому ядро здатне обробляти декілька операцій паралельно, що прискорює виконання програми.

Кожен порт мікроконтролера на основі ядра AVR складається з певного числа виводів, за допомогою яких можна здійснювати прийом та передачу цифрових сигналів так як даний контролер є 8 розрядним то він має 8 виводів. Виводи позначаються цифрами від 0 до 7p [23]. У мікроконтролерів даного типу усі виводи окрім живлення, по замовчуванню є виводами цифрових портів введення-виводу. Структурна схема мікроконтролерів сімейства AVR представлена на рисунку 2.2.

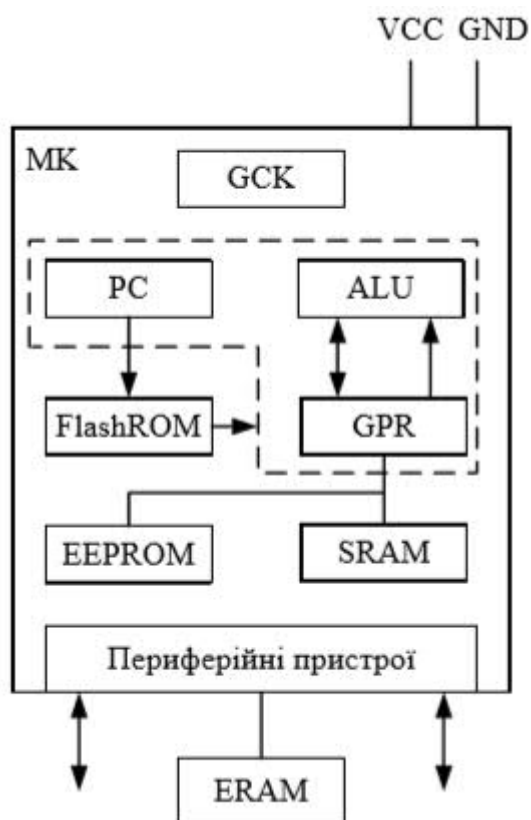


Рисунок 2.2 – Структурна схема мікроконтролера

Мікроконтролери орієнтовані на системи, які вимагають одночасно підвищеної продуктивності роботи та економічної роботи, а також використовують інтерфейси USB 2 [3]. Наявність 6-канального ШІМ-контролера з можливістю синхронізації каналів зовнішнім сигналом також

орієнтує ці можливості на застосування в системах керування електроприладами [27].

Мікроконтролер ATmega328 став серцем одноплатного комп'ютера Arduino uno, який забезпечує роботу пристрою послідовного виводу на дисплей. Саме завдяки своєму функціоналу, простотою та не великою ціною цей одноплатний комп'ютер здобув лідируючі позиції на ринку. І саме він є найкращим варіантом для використання в проекті.

## 2.2 Засоби візуалізації та методика налаштування системи

Візуалізація інформація дуже важливий фактор при розробці любого пристрою яким в подальшому мають користуватись люди. Оскільки для людей головне зручність користування і вони готові доплачувати за це гроші.

При розробці для візуалізації був використаний LCD дисплей 1602(рисунок 2.3) на основі контролера HD44780 розміром 2 стрічки по 16 символів, кожен символ може містити максимум 5 точок в ширину та 8 в висоту [6].

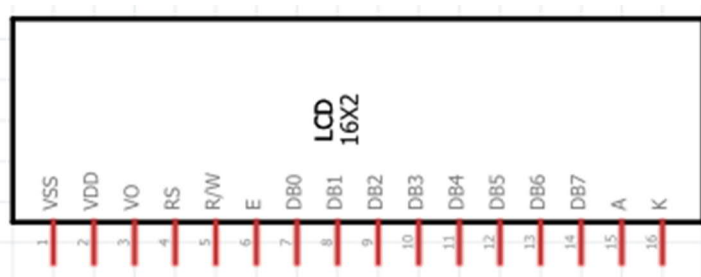


Рисунок 2.3 – Схема виводів LCD дисплея 1602

Також для зручності користування важливим буде використання якомога меншої кількості проводів, саме тому було використано послідовне підключення, а не паралельне(рисунок 2.4) при якому їх набагато більше.

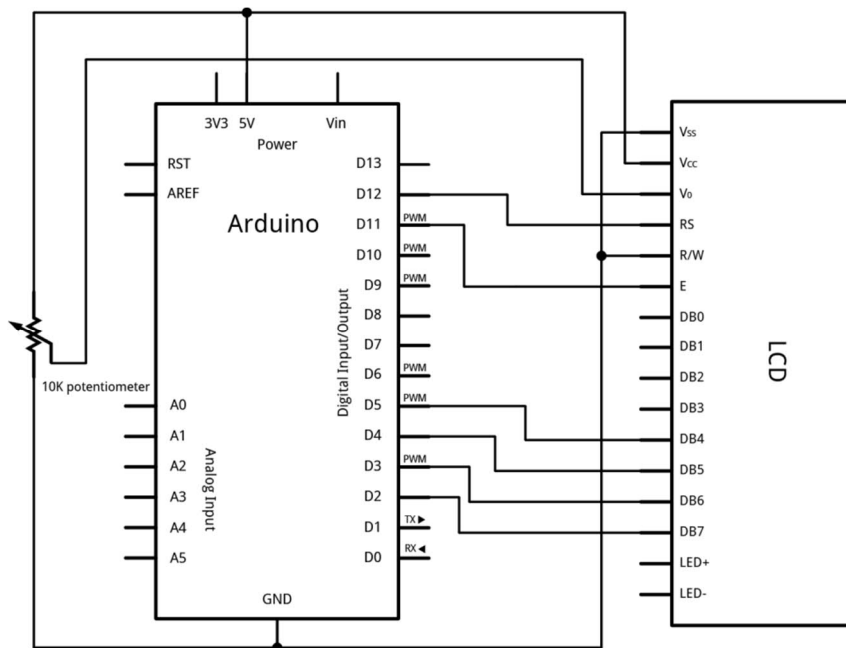


Рисунок 2.4 – Паралельне підключення Arduino до LCD дисплея

Проблема великої кількості проводів вирішується шляхом підключення через інтерфейс I2C тоді кількість проводів зменшується до 4, однак для підключення I2C потрібно щоб LCD був з послідовним інтерфейсом, такзваним Serial LCD тоді за допомогою мікросхеми MCP23017(рисунок 2.5) яка являє собою 8-бітний розширювач для I2C інтерфейсу [7]. Дана мікросхема не тільки дає можливість зменшити кількість проводів, а також використовуючи її можна підключити декілька пристроїв чи дисплеїв послідовно. Коли підключено більше одного дисплею спочатку звертаємось до його адреси, а пізніше передаємо на цей адрес пристрою необхідну інформацію для виводу.

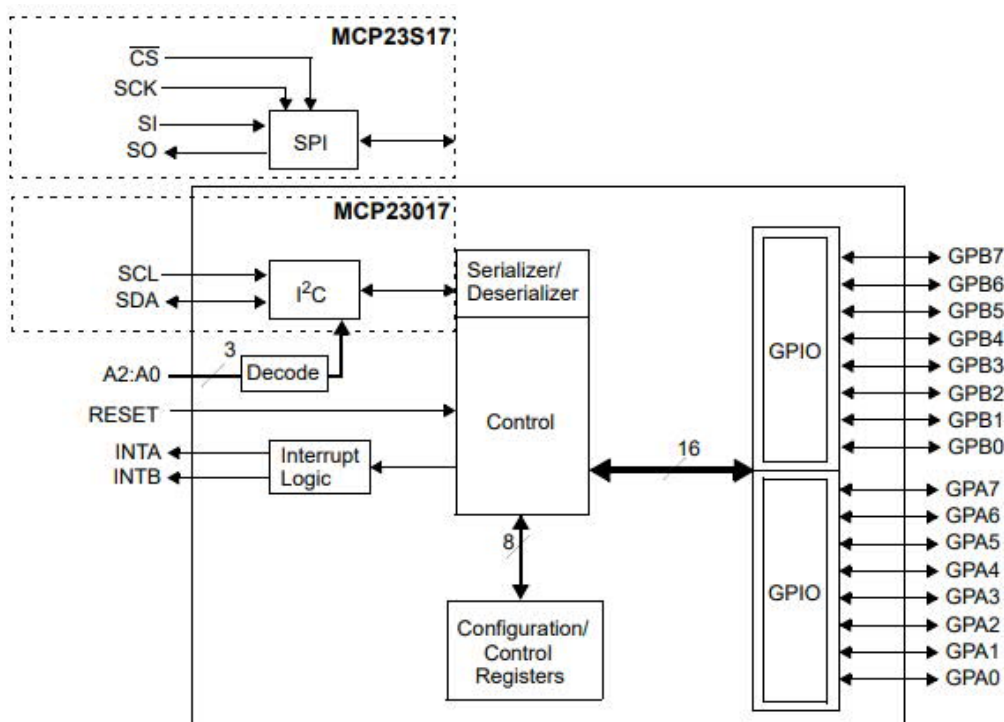


Рисунок 2.5– Структурна схема мікроконтролера MCP23017

Використовуючи I2C в розробці можна виділити такі переваги:

- для роботи потрібно лише дві лінії та живлення;
- зменшення часу розробки;
- можливість підключити велику кількість пристроїв;
- для керування великою кількістю пристроїв потрібен лише один мікроконтролер;
- висока ступінь надійності даних завдяки фільтру вбудованого в схему який приглушає спалахи;
- шина вже інтегрована в Arduino тому не потрібно розробляти додатковий інтерфейс.

В стандартному режимі забезпечується послідовна передача 8-бітних даних зі швидкістю в 100кб/с і до 400кб/с в швидкістному режимі роботи. Для реалізації обміну інформацією використовується два сигнали SDA – для передачі даних та SCL для синхронізації [5].



### 2.3 Алгоритм виводу інформації

Алгоритм – набір інструкцій, що описують порядок дій для досягнення певного результату. Набір інструкцій алгоритма зазвичай виконується послідовно, але також можна зробити його паралельним якщо система та мова програмування це дозволяють [11].

Під час виведення тексту на дисплей потрібно потурбуватись про ситуації коли текст більше вмістимості використовуваного дисплея, вмістимість вибраного LCD дисплея 16x2 тобто 32 символи, що не так вже й багато, особливо коли рахувати пробіл між словами окремим символом. В ситуаціях коли текста менше максимальної вмістимості ніяких складних маніпуляцій виконувати не потрібно. Однак коли символів більше потрібно придумати алгоритм виведення інформації. Розглянемо два варіанти виведення інформації яка не вміщається. Перший виведення інформації блоками в два рядки поступово, та зсувати інформацію знизу в верх. Перевагами такого методу є

- великий об'єм виводимої інформації.

Недоліками роботи даного алгоритму:

- складнота реалізації;
- незручність читання;
- велика кількість підводних каменів та помилок виконання.

Другий спосіб це виведення тексту за допомогою бігучої стрічки яка буде зміщуватися зліз на право показуючи нові символи. Перевагами даного методу є:

- зручність читання інформації;
- простота реалізації;
- можливість модифікації.

З недоліків:

- обмеження кількості символів.

Для реалізації було обрано другий алгоритм виводу інформації.

					БР.КСМ. 07090/15.00.00.000 ПЗ	29
Зм.	Арк.	№ докум.	Підпис	Дата		

Для коректної роботи та відображення було розроблено наступний алгоритм виводу інформації на дисплей – при отриманні стрічки типу string з текстом аналізуємо її, для початку перевіряємо довжину, якщо вона менше 16 символів то без змін виводимо текст, якщо більше 16, але менше 32 розділяємо по словах за допомогою символу «пробіл» який зазвичай знаходиться між словами, а пізніше рівномірно їх розділяємо у два рядки для виводу тексту, однак коли кількість символів більше 32, спочатку розділяємо по словах, а пізніше розподіляємо їх у два поля виводу, у перший рядок не більше 16 символів, в другий все решту буде виведено за допомогою бігучої стрічки. Алгоритм роботи розробляємо у Microsoft visio та в типі схеми обираємо «функціональна блок-схема» де є всі необхідні елементи для розробки блок схем.

#### 2.4 Вибір середовища автоматизованої розробки системи виведення інформації (Fritzing, Arduino IDE)

Для розробки схеми з Arduino було використано безкоштовне програмне забезпечення «Fritzing» яке містить весь необхідний функціонал для побудови схем у яких використовуюються однопроцесорні комп'ютери Arduino [25]. Функціонал програми доволі простий та інтуїтивний у використанні. При відкритті програми ми бачимо початкову сторінку з вікном вибору останніх проектів, перегляду новин програми (рисунок 2.7).

					БР.КСМ. 07090/15.00.00.000 ПЗ	30
Зм.	Арк.	№ докум.	Підпис	Дата		

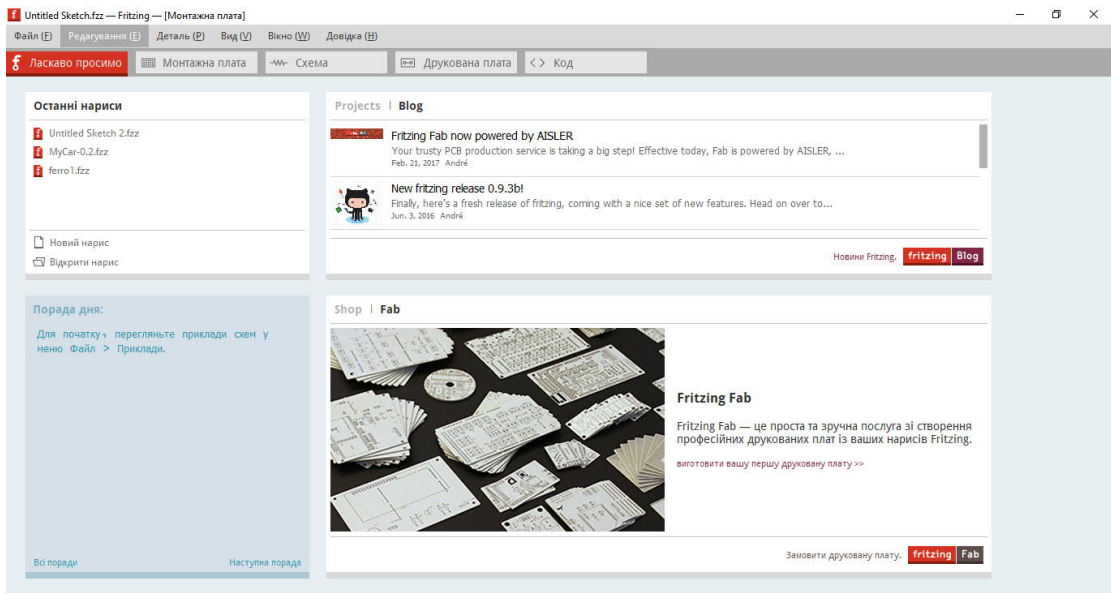


Рисунок 2.7 – Початкове вікно програми

Для пошуку нових деталей потрібно поставити галочку у меню «Вікно» біля пункта «Набір деталей» (рисунок 2.8), також знадобиться «Інспектор» для зміни характеристик деяких компонентів [25].

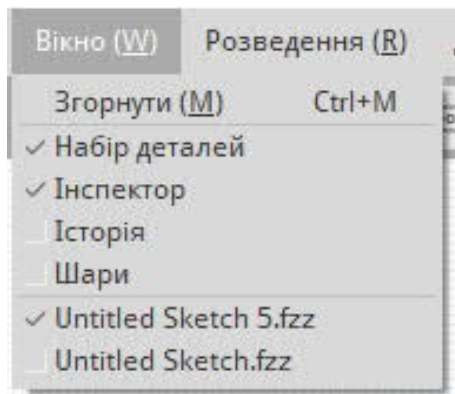


Рисунок 2.8 – Додання навігації для пошуку елементів

Також великою перевагою цього середовища над іншими – це можливість додавання нових елементів таких як драйверів чи інтерфейсів для підключення наприклад I2C який буде потрібен в подальшій розробці. Безліч різних елементів знаходиться в інтернеті у відкритому доступі, знайшовши необхідний достатньо лише у вільному просторі вікна компонентів натиснути

праву клавішу миші та імпортувати необхідний файл закачаний з інтернет  
рисунок 2.9.

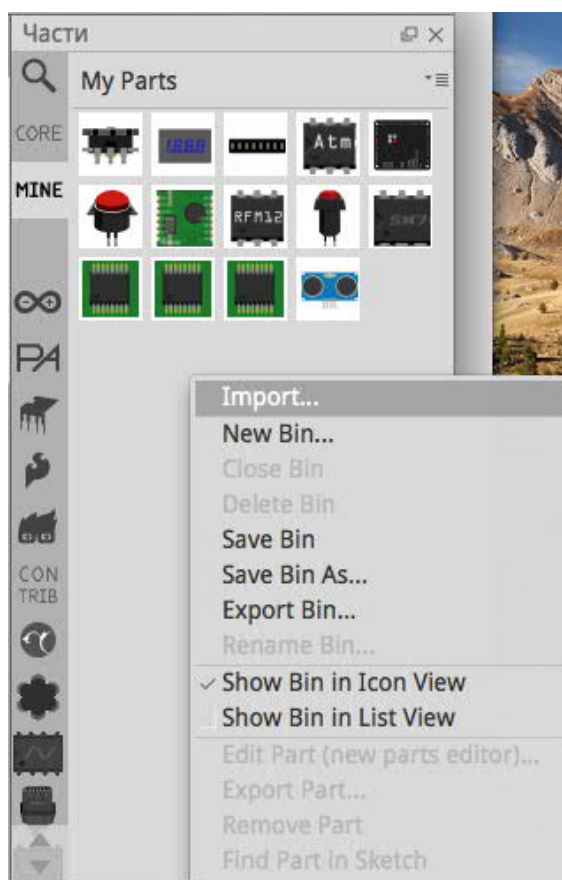


Рисунок 2.9 – Додавання нового компонента у Fritzing

Після розробки схеми у Fritzing необхідно запрограмувати мікроконтролер. Для цього використаємо середовище розробки Arduino IDE (рисунок 2.10).

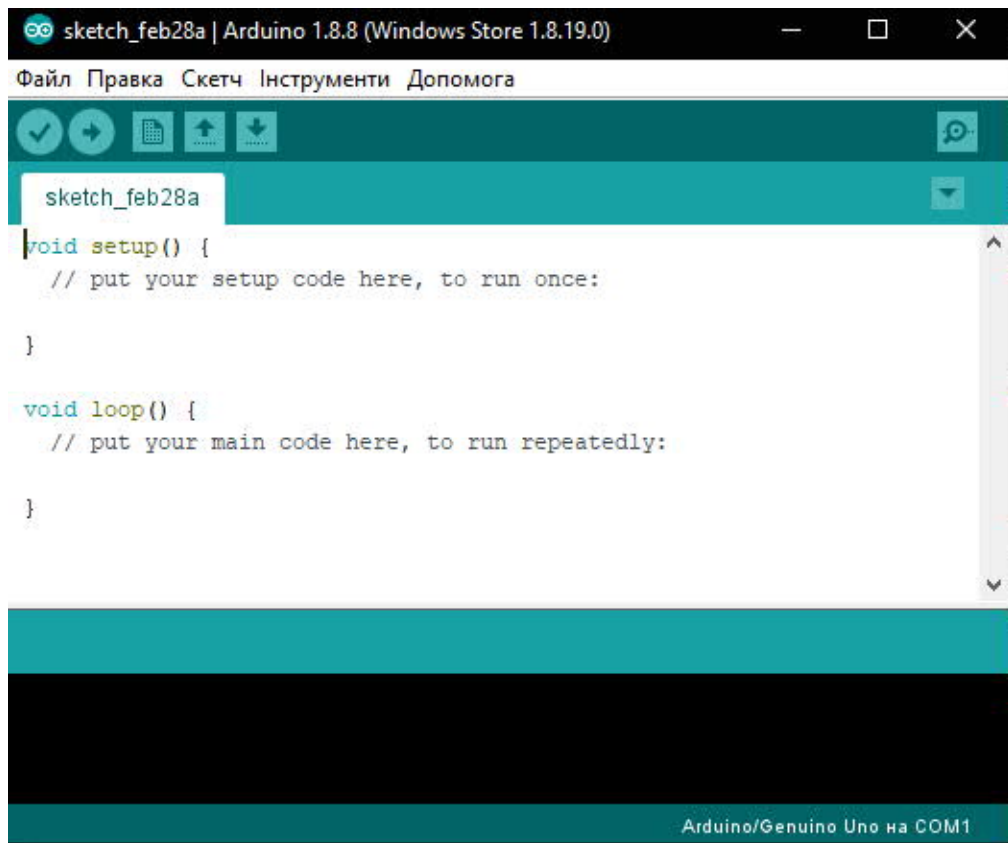


Рисунок 2.10 – Середовище розробки Arduino IDE

Програма, написана в середовищі Arduino, називається скетч. Скетч пишеться в редакторі ArduinoIDE. Під час збереження, експорту чи компіляції проекту в області повідомлень з'являється результат виконання чи помилка компіляції. Вікно виведення тексту (консоль) показує повідомлення Arduino, що включають повні звіти про помилки та іншу інформацію. Кнопки панелі інструментів дозволяють перевірити і записати програму, створити, відкрити і зберегти скетч, відкрити моніторинг послідовної шини [12].

Перед завантаженням скетчу потрібно задати необхідні параметри в меню Tools>Board і Tools>Serial Port. В ОС Windows порти можуть позначатися як COM1 або COM2 (для плати послідовної шини) або COM4, COM5, COM7 і вище (для плати USB) [26].

Після вибору порту і платформи необхідно натиснути кнопку завантаження на панелі інструментів або вибрати пункт меню File> Upload to I/O Board. Сучасні платформи Arduino перезавантажуються автоматично перед завантаженням. На старих платформах необхідно натиснути кнопку

перезавантаження. На більшості плат під час процесу будуть мигати світлодіоди RX і TX. Середовище розробки Arduino виведе повідомлення про закінчення завантаження або про помилки.

При завантаженні скетчу використовується Завантажувач (Bootloader) Arduino, невелика програма, що завантажується в мікроконтролер на платі. Вона дозволяє завантажувати програмний код без використання додаткових апаратних засобів. Завантажувач (Bootloader) активний протягом декількох секунд при перезавантаженні платформи і при завантаженні будь-якого з скетчів в мікроконтролер. Робота завантажувач (Bootloader) розпізнається по миганню світлодіода (13 пін) (напр . при перезавантаженні плати) [12].

					БР.КСМ. 07090/15.00.00.000 ПЗ	34
Зм.	Арк.	№ докум.	Підпис	Дата		

## 3 РЕАЛІЗАЦІЯ ПРИСТРОЮ ПОСЛІДОВНОГО ВИВЕДЕННЯ ІНФОРМАЦІЇ НА LCD ДИСПЛЕЙ

### 3.1 Апаратна реалізація пристрою

Для реалізації проекту будуть потрібні такі пристрої як паяльна станція, паяльник, мультиметр, а також розхідні матеріали такі, як припой, флюс, бензин галоша, щітки. Для тестування дисплея потрібно лише викрутку.

Основні характеристики пристрою:

- максимальна кількість відображуваних символів – 32;
- живлення – 7-12В, 120мА;
- максимальна тактова частота роботи пристрою – 16МГц;
- швидкість передачі даних по інтерфейсі I2C – 10-100кб/с.

Для початку складемо схему підключення в Frizing рисунок 3.1.

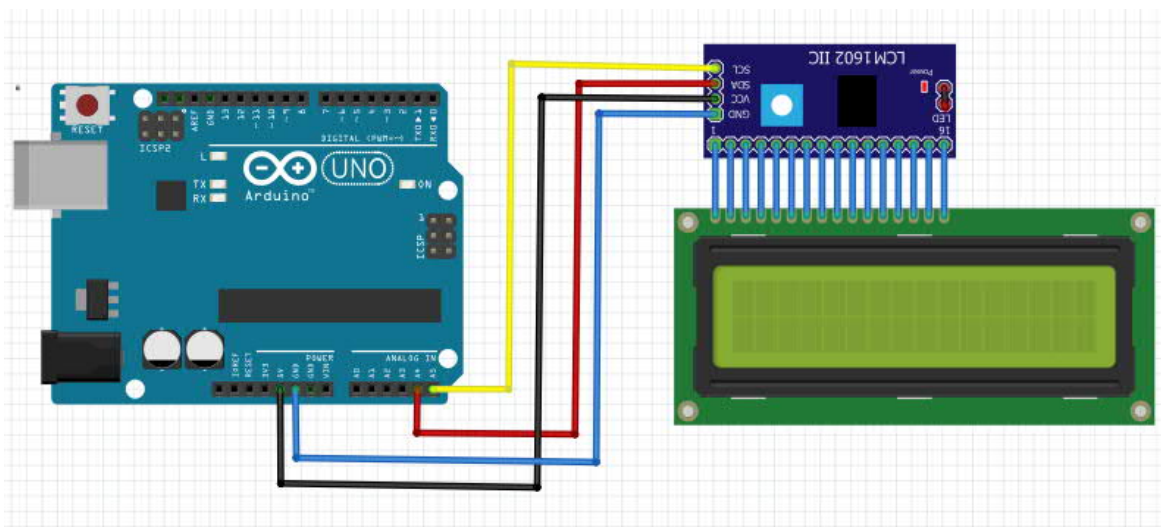


Рисунок 3.1 – Структурна схема пристрою

Першим кроком при зборці проекту потрібно припаяти інтерфейс I2C до LCD дисплея (рисунок 3.2), за допомогою паяльної пасти та станції припаюємо інтерфейс до плати дисплея, та за допомогою мультиметра продзвонюємо контакти як позначено на (рисунок 3.3) для перевірки підключення.

					БР.КСМ. 07090/15.00.00.000 ПЗ	
Зм.	Арк.	№ докум.	Підпис	Дата		35

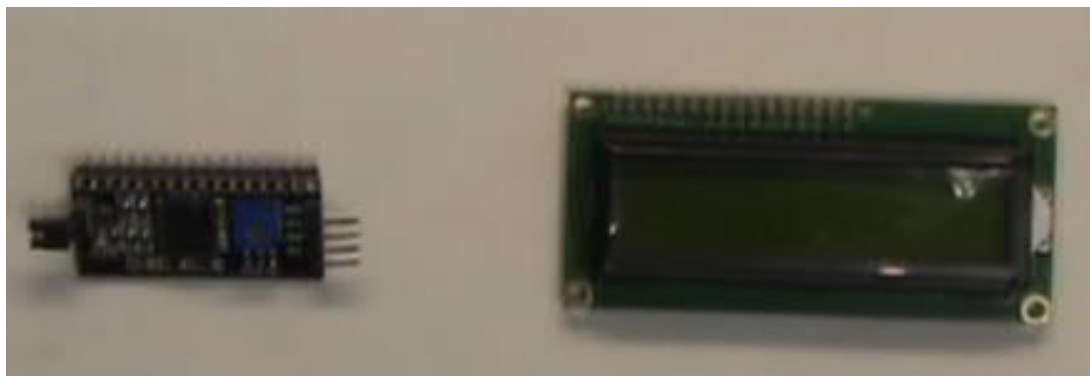


Рисунок 3.2 – Неприпаяні елементи

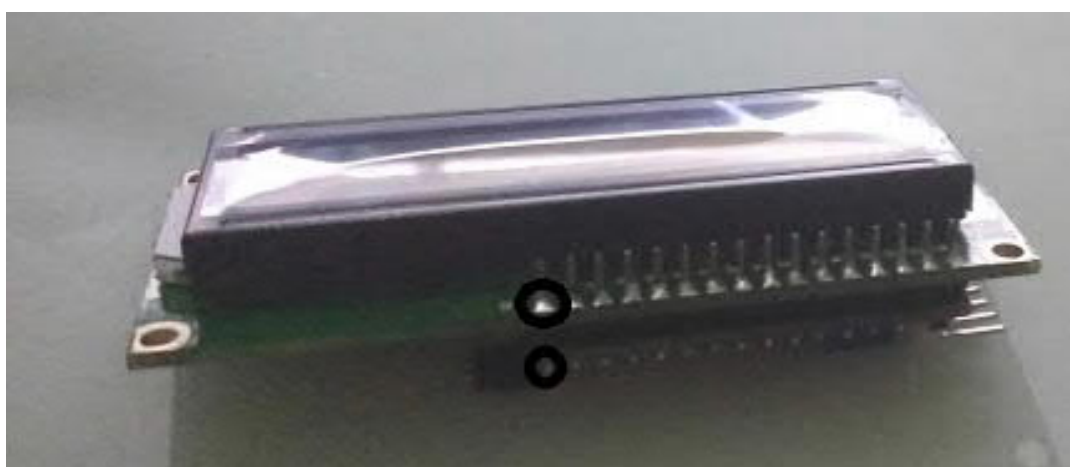


Рисунок 3.3 – Припаяні елементи

Підключення інтерфейсу до Arduino Uno буде відбуватися за допомогою проводів мама-тато, тому можна промивати флюс за допомогою бензину галоша. На рисунку 3.4 зображено повністю зібраний пристрій, виходи з інтерфейсу I2C підключаємо таким чином VCC- 5V, GND-GND, SDA-A4, SLC-A5 як зображено на рисунку 3.5 [17].





Рисунок 3.4 – Зібраний пристрій

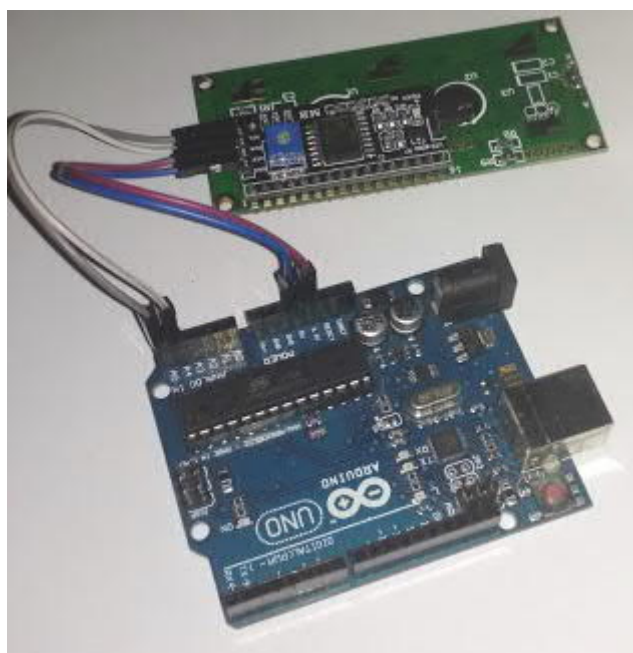


Рисунок 3.5 – Підключення LCD дисплея до платформи ArduinoUNO

Перед запуском обов'язково потрібно перевірити правильність підключення проводів щоб уникнути замикань та неправильної роботи пристрою.

## 3.2 Написання програмного коду роботи пристрою

Наступним кроком реалізації проекту буде написання коду роботи програми послідовного виводу інформації на дисплей в середовищі розробки Arduino IDE на мові C++. Для цього потрібно підключити ArduinoUNO до комп'ютера за допомогою кабеля USB type B–USB type A та відкривши середовище розробки Arduino IDE(рисунок 3.6) приступити до роботи.



Рисунок 3.6 – Середовище розробки Arduino IDE

Для початку потрібно підключити бібліотеку інтерфейсу I2C для роботи з дисплеєм яка знаходиться у вільному доступі мережі інтернет (рисунок 3.7) та встановимо тип дисплея та його розмір [2,1].

```
#include <LiquidCrystal_I2C.h>  
LiquidCrystal_I2C lcd(0x27,16,2);
```

Рисунок 3.7 – Підключення бібліотеки та встановлення типу дисплея

У блоці `void setup` ініціалізуємо дисплей за допомогою методу `lcd.init` та включаємо його підсвітку за допомогою `lcd.backlight()`, а у блоці коду `void loop` буде міститись код роботи програми рисунок 3.8 [2,20].

```
void setup()
{
  lcd.init();
  lcd.backlight();// Включаємо підсвітку
}
void loop()
```

Рисунок 3.8 – Включення підсвітки та ініціалізація

Ввід тексту наразі реалізовано тільки в середині програми в змінній `string` типу `char`, в подальшому можна буде налаштувати отримання інформації від іншого пристрою, датчика чи мережі інтернет [20]. Для обробки різних варіантів введеного тексту було використано оператор `if else`. Для пошуку точки розділення двох стрічок було використано цикл `for` та операцію «дикремент» та оператор умови `if` для знаходження символу «пробіл» при першому знаходженні виходимо з циклу за допомогою оператора `break` таким чином відділяємо весь вміст першого рядка дисплею рисунок 3.9 [4].

```
for(int i=16; i>=0;i--){
  if(string[i]==' '){
    a = i;
    break;
  }
}
for(int i=0; i<a;i++){//вивід першого рядка
  lcd.setCursor(i,0);
  lcd.print(string[i]);
}
```

Рисунок 3.9 – Пошук та вивід першого рядка на дисплей

Для виведення другого рядка в режимі «бігучої стрічки» спочатку потрібно відділити його від тексту який вже було виведено. Для цього створюємо дві нові стрічки одну пусу розміром який вже залишився, від довжину початкової стрічки віднімаємо порядковий номер кінця першого

рядка. Іншій стрічці присвоюємо першу та знаходимо символ початку другої стрічки, пізніше за допомогою оператора `strcpy` копіюємо весь потрібний текст в другу стрічку рисунок 3.10 [14].

```
char str2[roz-a];
char* ptr=string;
ptr=ptr+a+1;//відділяємо вже виведену частину тексту
strcpy(str2,ptr);
int razmer =strlen(str2);
```

Рисунок 3.10 – Виділяємо другий рядок тексту

Виконавши вказані вище операції можна приступити до виводу тексту на дисплей у режимі «бігучої стрічки». Для цього встановлюємо курсор на другий рядок та виводимо весь текст, за допомогою циклу поступово зсуваємо весь текст показуючи нові елементи, та якщо вже закінчилася стрічка продовжуємо зсувати текст. Крок зсуву 8 с та всі операції відображені на рисунку 3.11.

```
for (int in = 0; in <= razmer; in++)//алгоритм виведення бігучою стрічкою
{
  lcd.setCursor(0,1);
  lcd.print(str2);
  for(int j = 0; j < razmer; j++)
  {
    if (str2[j+1] == '\0')
    {
      str2[j] = ' ';
    }
    else
    str2[j] = str2[j+1];
  }
  delay(800);
}
```

Рисунок 3.11 – Виведення тексту у вигляді «бігучої стрічки»

В результаті роботи цього коду отримаємо блок схему яка зображена на рисунку Б.1. Тепер можна приступити до тестування роботи пристрою.

### 3.3 Налаштування та тестування

Налаштування відображення символів на дисплеї відбувається за допомогою прокручування викруткою змінного резистора який розміщений безпосередньо на платі I2C позначено на рисунку 3.12.



Рисунок 3.12 – Змінний резистор для регулювання дисплея

Так як текст для виведення прописується безпосередньо в коді програми залишається просто увімкнути живлення та перевірити правильність її роботи. Вивід тестового тексту зображено на рисунку 3.13.

					БР.КСМ. 07090/15.00.00.000 ПЗ	41
Зм.	Арк.	№ докум.	Підпис	Дата		



Рисунок 3.13 – Перевірка тестового виводу інформації

Виходячи з результатів тестування на LCD дисплей чітко відображає необхідний текст і повністю підходить для навчальних цілей. Також заявлені функції було протестовано та перевірено коректність виведення інформації та її максимальна кількість.

					БР.КСМ. 07090/15.00.00.000 ПЗ	42
Зм.	Арк.	№ докум.	Підпис	Дата		

## 4 ТЕХНІКО-ЕКОНОМІЧНИЙ РОЗДІЛ

### 4.1 Розрахунок витрат на розробку програмного забезпечення

Витрати на розробку і впровадження пристрою ( $K$ ) включають:

$$K = K_1 + K_2,$$

де  $K_1$  – витрати на розробку пристрою, грн.

$K_2$  – витрати на відлагодження і дослідну експлуатацію пристрою, грн.

Витрати на розробку пристрою включають:

- витрати на оплату праці розробників ( $B_{OP}$ );
- витрати на покупні вироби ( $П_B$ );
- витрати на придбання спец обладнання для проведення експериментальних робіт ( $O_{\bar{o}}$ );
- накладні витрати ( $H$ );
- інші витрати ( $I_B$ ).

#### 4.1.1 Розрахунок витрат на оплату праці

Витрати на оплату праці включають заробітну плату (ЗП) всіх категорій працівників, безпосередньо зайнятих на всіх етапах проектування. Розмір ЗП обчислюється на основі трудомісткості відповідних робіт у людино-днях та середньої ЗП відповідних категорій працівників.

У розробці проектного рішення задіяні наступні спеціалісти – розробники, а саме: керівник проекту; студент-дипломник; консультант техніко-економічного розділу.

					БР.КСМ. 07090/15.00.00.000 ПЗ	43
Зм.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.1 – Вихідні дані для розрахунку витрат на оплату праці

Посада виконавців	Місячний оклад, грн.
Керівник ДП, доцент	7293
Консультант техніко- економічного розділу, доцент	7293
Студент	1400

Витрати на оплату праці розробників проекту визначаються за формулою:

$$B_{OP} = \sum_{i=1}^N \sum_{j=1}^n n_{ij} * t_{ij} * c_{ij},$$

де  $n_{ij}$  – чисельність розробників і-ої спеціальності j-го тарифного розряду, осіб;

$t_{ij}$  – затрачений час на розробку проекту співробітником і-ої спеціальності j-го тарифного розряду, год;

$c_{ij}$  – годинна ставка працівника і-ої спеціальності j-го тарифного розряду, грн.

Середньо годинна ставка працівника може бути розрахована за формулою:

$$C_{ij} = \frac{c_{ij}^0 (1 + h)}{PЧ_i},$$

де  $C_{ij}$  – основна місячна заробітна плата розробника і-ої спеціальності j-го тарифного розряду, грн.;

					БР.КСМ. 07090/15.00.00.000 ПЗ	44
Зм.	Арк.	№ докум.	Підпис	Дата		



$h$  – коефіцієнт, що визначає розмір додаткової заробітної плати (при умові наявності доплат);

$РЧ_i$  – місячний фонд робочого часу працівника  $i$ -ої спеціальності  $j$ -го тарифного розряду, год. (приймаємо 168 год.).

Результати розрахунку записують до таблиці 4.2.

Таблиця 4.2 – Розрахунок витрат на оплату праці

Посада виконавців	Час розробки, год	Погодинна заробітна плата, грн/год.	Витрати на розробку, грн
Керівник ДП, доцент	18	63,89	1150
Консультант техніко-економічного розділу, доцент	2	63,8	127,6
Студент	100	8,3	830
Разом			2107.6

#### 4.1.2 Відрахування на соціальні заходи

Величну відрахувань у спеціальні державні фонди визначають у відсотковому співвідношенні від суми основної та додаткової заробітних плат. Згідно діючого нормативного законодавства сума відрахувань у спеціальні державні фонди складає 20,5 % від суми заробітної плати:

$$B_{\phi} = \frac{20,5}{100} * 2107,6 = 432 \text{ грн.}$$

#### 4.1.3 Розрахунок витрат на матеріали та комплектуючі

У таблиці 4.3 наведений перелік купованих виробів і розраховані витрати на них.

Таблиця 4.3– Розрахунок витрат на матеріали та комплектуючі

Найменування купованих виробів	Одиниця виміру	Ціна, грн	Кількість купованих виробів	Сума, грн	Транспортні витрати (10% від суми)	Загальна сума, грн
Arduino Uno Rev3 з кабелем USB	шт	209	1	209	20.9	229.9
LCD дисплей 20x4 з синьою підсвіткою	шт	157	1	157	15.7	172.7
Проводи мама-тато	шт	1	5	5	0,5	10,5
Живлення 9В	шт	90	1	90	9	99
Флюс	уп	125	1	125	12,5	137,5
Припой	уп	20	1	20	2	22
Розчинник	уп	15	1	15	1,5	16,5
Разом						688,1

#### 4.1.4 Витрати на використання комп'ютерної техніки

Витрати на використання комп'ютерної техніки включають витрати на амортизацію комп'ютерної техніки, витрати на користування програмним забезпеченням, витрати на електроенергію, що споживається комп'ютером. За даними обчислювального центру ТНЕУ для комп'ютера типу IBM PC/ATX вартість години роботи становить 5.2 грн. Середній щоденний час роботи на комп'ютері – 3 години. Розрахунок витрат на використання комп'ютерної техніки приведений в таблиці 4.4.

Таблиця 4.4– Розрахунок витрат на використання комп'ютерної техніки

Назва етапів робіт, при виконанні яких використовується комп'ютер	Час використання комп'ютера, год.	Витрати на використання комп'ютера грн.
Проведення досліджень та оформлення їх результатів	60	312
Оформлення техніко-економічного розділу	5	26
Оформлення ДП	25	130
Разом	90	520

#### 4.1.5 Накладні витрати

Накладні витрати проектних організацій включають три групи видатків: витрати на управління, загальногосподарські витрати, невиробничі витрати. Вони розраховуються за встановленими відсотками до витрат на оплату праці. Середньостатистичний відсоток накладних витрат прийmemo 150% від заробітної плати:

$$H = 1,5 * 2107,6 = 3161,4 \text{ грн.}$$

#### 4.1.6 Інші витрати

Інші витрати є витратами, які не враховані в попередніх статтях. Вони становлять 10% від заробітної плати:

$$I = 2107,6 * 0,1 = 210,7 \text{ грн.}$$

					БР.КСМ. 07090/15.00.00.000 ПЗ	47
Зм.	Арк.	№ докум.	Підпис	Дата		

Витрати на розробку програмного забезпечення складають:

$$K_1 = B_{ОП} + B_{\Phi} + B_{ГВ} + H + I,$$

$$K_1 = 2107,6 + 432 + 688,1 + 3161,4 + 210,7 = 6599,8 \text{ грн.}$$

Витрати на відлагодження і дослідну експлуатацію програмного продукту визначаємо за формулою:

$$K_2 = S_{м.г.} * t_{від.},$$

де  $S_{м.г.}$  – вартість однієї машино-години роботи ПК, грн./год.

$t_{від}$  – комп'ютерний час, витрачений на відлагодження і дослідну експлуатацію створеного програмного продукту, год.

Загальна кількість днів роботи на комп'ютері дорівнює 30 днів. Середній щоденний час роботи на комп'ютері – 3 години. Вартість години роботи комп'ютера дорівнює 5.2 грн. Тому

$$K_2 = 5,2 * 90 = 520 \text{ грн.}$$

На основі отриманих даних складаємо кошторис витрат на розробку програмного забезпечення.

Таблиця 4.5 – Кошторис витрат на розробку програмного забезпечення

Найменування витрат	Сума витрат, грн.
1	2
Витрати на оплату праці	2107.6

Продовження таблиці 4.5

1	2
Відрахування у спеціальні державні фонди	432
Витрати на куповані вироби	688,1
Накладні витрати	3161,4
Інші витрати	210,7
Витрати на відлагодження і дослідну експлуатацію програмного продукту	520
Разом	7119,8

#### 4.2 Визначення експлуатаційних витрат

Для оцінки економічної ефективності розроблюваного програмного продукту слід порівняти його з аналогом, тобто існуючим програмним забезпеченням ідентичного функціонального призначення.

Експлуатаційні одноразові витрати по програмному забезпеченню і аналогу включають вартість підготовки даних і вартість роботи комп'ютера (за час дії програми):

$$E_{II} = E_{1II} + E_{2II},$$

де  $E_{II}$  – одноразові експлуатаційні витрати на пристрій (аналог), грн.;

$E_{1II}$  – вартість підготовки даних для експлуатації пристрою (аналог), грн.;

$E_{2II}$  – вартість роботи комп'ютера для виконання проектного рішення (аналог), грн.

Річні експлуатаційні витрати  $B_{EII}$  визначаються за формулою:

$$B_{EП} = E_{П} * N_{П},$$

де  $N_{П}$  – періодичність експлуатації пристрою (аналогу), раз/рік.

Таблиця 4.6 – Розрахунок витрат на підготовку даних та реалізацію програмного забезпечення на комп'ютері.

Час роботи співробітників	Середньогодинна заробітна плата, грн./год.	Витрати , грн.
Проектне рішення		
1	8,3	8,3
Аналог		
2	7,15	14,3

Витрати на експлуатацію комп'ютера визначається за формулою:

$$E_{2П} = t * S_{МГ},$$

де  $t$  – витрати машинного часу для реалізації проектного рішення (аналогу), год.

$$E_{П} = 1 * 10,55 = 8,3 \text{ грн.}; E_{А} = 2 * 7,15 = 14,3 \text{ грн.}$$

$$B_{EП} = 8,3 * 252 = 2091,6 \text{ грн.}; B_{EA} = 13,3 * 252 = 3351,6 \text{ грн.}$$

### 4.3 Розрахунок ціни споживання проектного рішення

Ціна споживання – це витрати на придбання і експлуатацію проектного рішення за весь строк його служби:

$$Ц_{C(\Pi)} = Ц_{\Pi} + B_{(E)NPV},$$

де  $Ц_{\Pi}$  – ціна придбання проектного рішення, грн.:

$$Ц_{\Pi} = K(1 + \frac{\Pi_P}{100}) + K_0 + K_K,$$

де  $K$  – кошторисна вартість;

$\Pi_P$  – рентабельність;

$K_0$  – витрати на прив'язку та освоєння проектного рішення на конкретному об'єкті, грн.;

$K_K$  – витрати на доукомплектування технічних засобів на об'єкті, грн.;

$$Ц_{\Pi} = 7119,8 * (1 + 0,3) = 9255,7 \text{ грн.}$$

$$Ц_{\Pi A} = 7563,2 * (1 + 0,3) = 9832,2 \text{ грн.}$$

Вартість витрат на експлуатацію проектного рішення (за весь час його експлуатації), грн.:

					БР.КСМ. 07090/15.00.00.000 ПЗ	51
Зм.	Арк.	№ докум.	Підпис	Дата		

$$B_{епрв} = \sum_{t=0}^T \frac{B_{eП}}{(1+R)^t},$$

де  $B_{eП}$  – річні експлуатаційні витрати, грн.;

$T$  – строк служби проектного рішення, років;

$R$  – річна ставка проценту банку.

$$B_{епрв} = \sum_{t=1}^5 \frac{2091,6}{(1+0,05)^t} = 9055,5 \text{ грн.}$$

$$B_{епрв} = \sum_{t=1}^5 \frac{3351,6}{(1+0,05)^t} = 14467,4 \text{ грн.}$$

Тоді ціна споживання проектного рішення дорівнюватиме:

$$Ц_{СП} = 9255,7 + 9055,5 = 18311,2 \text{ грн.}$$

Аналогічно визначається ціна споживання для аналогу:

$$Ц_{СА} = 9832,2 + 14467,4 = 24299,6 \text{ грн.}$$

#### 4.4 Визначення показників економічної ефективності

Економічний ефект в сфері проектування рішення:

$$E_{ПР} = Ц_{А} - Ц_{П},$$

					БР.КСМ. 07090/15.00.00.000 ПЗ	52
Зм.	Арк.	№ докум.	Підпис	Дата		



$$E_{\text{ІПР}} = 9832,2 - 9255,6 = 576,6 \text{ грн.}$$

Річний економічний ефект в сфері експлуатації:

$$E_{\text{КС}} = B_{\text{ЕА}} - B_{\text{ЕП}},$$

$$E_{\text{КС}} = 3351,6 - 2091,6 = 1260$$

Додатковий економічний ефект у сфері експлуатації:

$$\Delta E_{\text{екс}} = \sum_{t=1}^T E_{\text{екс}} (1 + R)^{T-t},$$

$$\Delta E_{\text{екс}} = \sum_{t=1}^5 1260(1 + 0,05)^{5-t} = 6962,3 \text{ грн.}$$

Сумарний ефект складає:

$$E = E_{\text{ІПР}} + \Delta E_{\text{екс}} = 576,6 + 6962,3 = 7438,9 \text{ грн.}$$

Таблиця 4.7 – Показники економічної ефективності проектного рішення

Найменування	Одиниці вимірювання	Значення показників	
		Базовий варіант	Новий варіант
1	2	3	4
Капітальні вкладення	грн.	–	7119,8
Ціна придбання	грн.	9832,2	9255,7
Річні експлуатаційні витрати	грн.	14467,4	9055,5

Ціна споживання	грн.	24299,6	18311,2
Економічний ефект в сфері проектування	грн.	–	576,6

Продовження таблиці 4.7

1	2	3	4
Економічний ефект в сфері експлуатації	грн.	–	1260
Додатковий ефект в сфері експлуатації	грн.	–	6962,3
Сумарний ефект	грн.	7438,9	

Після проведення математичних обчислень було встановлено, що проектне рішення є оправданим та економічно вигідним. Також вище наведено всі обрахунки які обґрунтовують економічну вигоду від впровадження.

## ВИСНОВКИ

В результаті виконання дипломного проекту на тему «Послідовний вивід інформації на LCD дисплей» отримано наступні результати.

1. Аналіз сучасних тенденцій застосування мікроконтролерів показав їх важливість при вирішенні задач автоматизації функцій невеликих об'єктів.

2. Особливістю функціонування мікроконтролерної системи є важливість візуалізації результатів управління та контролю в місці застосування мікроконтролерної системи.

3. Широка номенклатура мкропроцесорів та дисплеїв різних розмірів та цінкових діапазонів створюють можливості для розробки недорогих та багатофункціональних мікроконтролерних пристроїв.

4. Здійснено постановку задачі по створенню пристрою візуалізації інформації на базі мікроконтролера Arduino Uno.

5. Вибрано та проаналізовано особливості роботи LCD дисплеїв.

6. З метою реалізації задачі розроблено структурно-функціональну схему пристрою візуалізації інформації на базі інтерфейсу I2C.

7. Обгрунтовано та розроблено алгоритм послідовного виведення інформації на диспей.

8. В програмному середовищі Arduino IDE розроблено скетч виведення інформації на LCD-дисплей.

9. Техніко-економічне обгрунтування показало економічну доцільність розробки системи візуалізації показів мікроконтролера.

					БР.КСМ. 07090/15.00.00.000 ПЗ	55
Зм.	Арк.	№ докум.	Підпис	Дата		

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Arduino Language Reference: веб-сайт.URL:  
<http://www.arduino.cc/en/Reference/HomePage> (дата звернення: 07.03.2019).
2. Arduino LiquidCrystal I2C: веб-сайт. URL:  
<https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library> (дата звернення: 25.02.2019).
3. ATmega128 Datasheet – Atmel Corporation: веб-сайт.URL:  
<http://www.gaw.ru/pdf/Atmel/AVR/atmega128.pdf> (дата звернення: 20.05.2019).
4. C++ Сайт документація бібліотеки: веб-сайт. URL:  
<http://www.cplusplus.com/reference/cstring/?kw=string.h> (дата звернення: 20.04.2019).
5. I2C інтерфейс: веб-сайт. URL: [http://itt-ltd.com/reference/ref\\_i2c.html](http://itt-ltd.com/reference/ref_i2c.html)  
 (дата звернення: 15.05.2019).
6. LCD Datasheet: веб сайт. URL: <http://www.alldatasheet.com/datasheet-pdf/1358370/LCD.html> (дата звернення: 23.02.2019).
7. MCP23017 Datasheet: веб-сайт. URL:  
<http://ww1.microchip.com/downloads/en/devicedoc/20001952c.pdf> (дата звернення: 13.04.2019).
8. Баранов В. Н. Применение микроконтроллеров AVR. / Додэка-XXI, 2004. 288 с
9. Баранов В.Н. Применение микроконтроллеров AVR: схемы, алгоритмы, программы. / Издательство: Додэка, 2004. 289 с.
10. Белов, А. В. Конструювання пристроїв на мікроконтролерах. / СПБ Наука і Техніка, 2005. С. 9–149.
11. Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих / пер. с англ. Е. Матвеев. Питер, 2017. С 20-43.

					БР.КСМ. 07090/15.00.00.000 ПЗ	
Зм.	Арк.	№ докум.	Підпис	Дата		56

12. Гоноровський, І. С. Ардуїно і програмування: Підручник для вузів. / Програмування, 2012. Т.4. 512 с.

13. Гребнев В. В. Микроконтроллеры семейства AVR фирмы Atmel. / Издательство: РадиоСофт, 2002. С. 110–138.

14. Иванов Ю. І. Мікропроцесорні пристрої систем управління: Навчальний посібник / Видавництво ТРТУ, 2005. С. 135–180.

15. Каштелян І.В, Данило А.М. Мікроконтролерні системи – складова розподілених систем автоматизації: зб. матеріалів доп. учасн. наук.-практ. конф. Тернопіль: Тернопіль 2019. С.16.

16. Коффон Д. Технические средства микропроцессорных систем; Практический курс. / Пер. с англ. Д. Коффон. Мир, 1983. 178с

17. Майоров В. Г., Гаврилов А. И. Практический курс программирования микропроцессорных систем / Машиностроение, 2009. 272с.

18. Методичні вказівки до оформлення курсових проектів, звітів про проходження практики, випускних кваліфікаційних робіт для студентів спеціальності «Комп'ютерна інженерія» /І.В.Гураль, Л.О.Дубчак / Під ред. О.М,Березького . Тернопіль: ТНЕУ, 2019. 33с.

19. Методичні рекомендації до виконання дипломного проекту з освітньо-кваліфікаційного рівня “Бакалавр” напряму підготовки 6.050102 «Комп'ютерна інженерія» фахового спрямування «Комп'ютерні системи та мережі» / О.М. Березький, Л.О.Дубчак, Р.Б. Трембач, Г.М. Мельник, Ю.М. Батько, С.В. Івасєв / Під ред. О.М. Березького. - Тернопіль: ТНЕУ, 2013.–65с.

20. Монк С. Програмуємо Arduino. Професійна робота со скетчами / Питер, 2015. С. 25-55.

21. Новицкий А. Синхронный последовательный интерфейс SPI в микроконтроллерах от «А до Я» и его реализация на примере ADuC70xx фирмы analog devices: Компоненты и технологии. 2009. N 3.С. 37-48.

					БР.КСМ. 07090/15.00.00.000 ПЗ	
Зм.	Арк.	№ докум.	Підпис	Дата		57

22. Паздрій І.Р. Методичні вказівки до написання техніко-економічного розділу дипломних проектів освітньо-кваліфікаційного рівня «бакалавр» – Тернопіль: Економічна думка 2014.- 36 с.

23. Пархоменко Д. А. Розробка радіоелектронних схем на основі мікроконтролерів (на прикладі AVR мікроконтролерів фірми Atmel): методичний посібник до курсу "Проектування радіоелектронних схем" для студентів радіофізичного факультету / Радіофізичний факультет КНУ ім. Тараса Шевченка, 2013. С. 35–74.

24. Послідовний інтерфейс SPI: веб-сайт. URL: <http://www.gaw.ru/html.cgi/txt/interface/spi/index.htm> (дата звернення: 27.04.2019).

25. Сайт програмного середовища Fritzing: веб сайт. URL: <http://fritzing.org/learning/get-started/> (дата звернення 21.03.2019).

26. Сайт програмного середовища компанії Arduino: веб-сайт. URL:<https://www.arduino.cc> (дата звернення: 23.02.2019).

27. Соммер, У. Программирование микроконтроллерных плат Arduino/Freduino / пер. с нем. Букирева В. Петербург, 2012. 241 с.