

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**  
Факультет комп'ютерних наук, фізики та математики  
Кафедра комп'ютерних наук та програмної інженерії

Методи і технології програмування автоматів з продажу рідин

**Кваліфікаційна робота (проект)**  
на здобуття ступеня вищої освіти «магістр»

Виконав: здобувач 2 курсу 241 групи

Спеціальність: 121 Інженерія програмного забезпечення

Освітньо-професійна програма: інженер-програміст

Романенко Андрій Павлович

Керівник: к. фіз.-мат. Н., доцент, Кравцов Г.М.

Рецензент: вчитель інф. Херсонського фіз.-тех. ліцею Тарасюк А.О.

Івано-Франківськ – 2024

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ І ТЕРМІНІВ.....	4
ВСТУП.....	5
РОЗДІЛ 1. Аналіз літератури та технологій.....	9
1.1 Огляд сучасних методів продажу води через автомати.....	9
1.2 Аналіз програмних засобів управління автоматами.....	11
РОЗДІЛ 2. Вимоги до програмного забезпечення.....	13
2.1 Архітектура системи автомату.....	13
2.2 Опис функціональних модулів системи.....	14
2.3 Вибір апаратного забезпечення.....	17
2.4 Алгоритми роботи автомату.....	21
РОЗДІЛ 3. Опис алгоритмів реалізації вибору продукту.....	25
3.1 Розрахунок вартості та контроль оплати.....	25
3.2 Алгоритми дозування та видачі продукту.....	27
3.3 Управління сенсорами та акторами.....	31
3.4 Програмування та тестування системи.....	33
РОЗДІЛ 4. Вибір мови програмування та інструментів розробки.....	36
4.1 Основні етапи програмування системи.....	36
4.2 Тестування програмного забезпечення та моделювання роботи автомата.....	39
РОЗДІЛ 5. Оцінка вартості розробки та впровадження.....	42
5.1 Оцінка економічної ефективності системи.....	42
5.2 Порівняльний аналіз з аналогічними рішеннями на ринку.....	45

5.3 Перспективи розвитку та вдосконалення.....	48
ВИСНОВКИ.....	52
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	57
Додаток А. Загальний алгоритм опрацювання системи автомату.....	59
Додаток Б. Алгоритм опрацювання кнопки користувача.....	61
Додаток В. Алгоритм опрацювання платіжної системи.....	62
Додаток Г. Програмний код.....	63

## ПЕРЕЛІК СКОРОЧЕНЬ І ТЕРМІНІВ

Вендінг (від англ. "vending") — це автоматизована торгівля через спеціальні торгові автомати, які здійснюють продаж товарів або послуг без участі продавця. Вендінгові автомати поширені в різних сферах і можуть пропонувати широкий асортимент товарів, таких як напої, їжа, квитки, засоби особистої гігієни тощо.

Валідатор — це пристрій або програма, що перевіряє правильність або дійсність певної інформації чи даних. У контексті вендінгових автоматів, валідатором зазвичай називають пристрій для перевірки і визнання платіжних засобів, таких як купюри, монети, картки чи інші платіжні системи.

NFC (Near Field Communication) — це технологія бездротового зв'язку малого радіуса дії, яка дозволяє обмінюватися даними між пристроями на відстані до 10 сантиметрів. Технологія NFC дозволяє зчитувати інформацію з міток або здійснювати передачу даних між пристроями, що робить її популярною у вендінгових автоматах, смартфонах та інших платіжних системах.

ПЗ — це скорочення від "програмне забезпечення". Це загальний термін, який використовується для позначення всіх видів програм, додатків і систем, що виконують різні функції на комп'ютерах, мобільних пристроях та інших електронних пристроях. Програмне забезпечення може включати операційні системи, прикладні програми, системи управління базами даних, ігри, утиліти та багато інших типів програм.

## ВСТУП

На сьогоднішній день питання автоматизації процесів торгівлі товарами та послугами набуло значної актуальності. Це стосується не лише класичних рішень у сфері роздрібної торгівлі, але й інноваційних систем, що дозволяють швидко і зручно забезпечувати населення необхідними ресурсами. Однією з таких систем є вендінгові торгові автомати, які здійснюють продаж рідин, зокрема питної води, соків, молока, напоїв та інших продуктів. Ці автомати стають не лише зручними та доступними для споживачів, але й грають важливу роль у вирішенні проблем екологічної та економічної ефективності.

В даній роботі сконцентруємо увагу на окремому виді вендінгових машин – автомати з продажу води, оскільки це є надзвичайно актуальною з кількох причин:

1) Проблема доступу до якісної питної води. У багатьох регіонах світу, включно з деякими частинами України, питання якості питної води стоїть дуже гостро. Центральне водопостачання часто не забезпечує належної якості води, а купівля бутильованої води може бути дорогою та незручною для багатьох людей. Вендінгові автомати, які продають очищену воду, дозволяють забезпечити легкий доступ до якісної питної води без необхідності витратити великі кошти на упаковку та доставку.

2) Екологічний аспект. Продаж води через автомати сприяє зниженню використання одноразової пластикової тари, яка є одним з найбільших джерел забруднення довкілля. Заохочення споживачів використовувати власні багаторазові пляшки або тару для повторного наповнення через автомати сприяє зменшенню обсягу пластикових відходів, що накопичуються на планеті. Це позитивно впливає на довкілля і водночас популяризує концепцію відповідального споживання.

3) Ефективність та зручність. Вендінгові автомати з продажу води забезпечують зручність для споживачів, оскільки дозволяють купувати воду в будь-який час доби, без необхідності відвідувати магазини або супермаркети. Це особливо важливо у міських умовах, де ритм життя є швидким, а доступ до магазинів може бути обмеженим вночі або у святкові дні.

4) Скорочення витрат на логістику та транспортування. Традиційна модель розповсюдження бутильованої води вимагає значних витрат на транспортування, зберігання та обробку пластикових пляшок. Вендінгові автомати, розташовані безпосередньо біля споживача, скорочують ці витрати, роблячи процес постачання води економічно вигіднішим.

4) Глобальна тенденція до сталого розвитку. В сучасному світі значний акцент робиться на сталому розвитку, і вендінгові автомати з продажу води відповідають цій концепції. Вони дозволяють зменшити негативний вплив на навколишнє середовище, одночасно забезпечуючи економічну вигоду для бізнесу та доступність для споживачів. Ця тема є важливою частиною переходу до екологічно безпечних рішень у торгівлі товарами та послугами.

Таким чином, виокремлення теми автоматів з продажу води обумовлено, як соціальними, так і екологічними аспектами, що робить це питання надзвичайно актуальним та важливим для дослідження в сучасних умовах.

Технологічні досягнення останніх років суттєво змінили підхід до програмування вендінгових автоматів. Сучасні програмні рішення для автоматів включають інтеграцію різних методів оплати, таких як банківські картки та мобільні платежі, що значно розширює можливості використання систем для різних категорій споживачів. Також все більшого поширення набувають системи віддаленого моніторингу та

управління, які дозволяють операторам автоматів контролювати їх роботу у реальному часі, прогнозувати технічні несправності, оновлювати асортимент продуктів та управляти логістичними процесами. Це підвищує ефективність та знижує витрати на обслуговування таких систем.

Важливою складовою роботи з автоматами є системи безпеки. Програмне забезпечення повинно забезпечувати надійний захист від збоїв у роботі автоматів, гарантувати безпеку платежів та забезпечувати конфіденційність особистих даних користувачів. Крім того, програмне забезпечення автоматів повинно бути адаптованим до різних умов експлуатації, що включає роботу у різних кліматичних зонах, витримування коливань електричної напруги та інші зовнішні фактори.

Метою цієї дипломної роботи є дослідження сучасних методів і технологій програмування вендінгових торгових автоматів для продажу рідин. В рамках роботи буде проведено аналіз існуючих технічних рішень, вивчено перспективи розвитку галузі та запропоновано вдосконалення програмного забезпечення автоматів для підвищення їх ефективності, безпеки та екологічної доцільності. Особлива увага буде приділена аналізу та практичному застосуванню протоколів платіжних систем, керуючих пристроїв та інтерфейсів користувача, що є ключовими елементами вендінгових торгових автоматів. Зокрема, буде досліджено, як інтегрувати різні платіжні системи, такі як монетоприймачі та валідатори, також буде розглянута можливість інтегрування оплати банківськими картками, мобільними платежами та застосування NFC-технології, з програмним забезпеченням автоматів для забезпечення безпеки транзакцій та зручності користувачів.

Також буде проведений аналіз керуючих пристроїв, включаючи

мікроконтролери, датчики та приводи, що відповідають за розподіл рідин та взаємодію з іншими компонентами системи. Це дозволить створити ефективну архітектуру управління, здатну забезпечити точне дозування рідини, контроль за якістю води та підтримку працездатності автомату в реальному часі.

Окрему увагу буде приділено інтерфейсам користувача, що забезпечують легкий і зрозумілий процес взаємодії з автоматом. Розробка інтуїтивного і простого інтерфейсу сприятиме покращенню користувацького досвіду та збільшенню ефективності використання автоматів.

Таким чином, робота спрямована на комплексний підхід до вирішення проблеми, включаючи як технічну складову, так і зручність та безпеку для кінцевих користувачів.

Актуальність теми дослідження зумовлена зростанням попиту на автоматизовані торгові системи, які забезпечують безперебійну роботу у різних умовах, зменшують витрати на обслуговування та сприяють екологічному відповідальному споживанню. Умови сучасного ринку потребують високого рівня автоматизації процесів торгівлі, а також впровадження нових екологічних рішень. Використання таких автоматів сприятиме мінімізації утворення побутових відходів, зменшенню негативного впливу на довкілля та покращенню доступу до якісних рідин для широкого кола споживачів.



## РОЗДІЛ 1. Аналіз літератури та технологій

### 1.1 Огляд сучасних методів продажу води через автомати

Автоматизовані системи продажу води стають дедалі популярнішими у світі завдяки їх зручності, доступності та економічній ефективності. Ці системи інтегрують сучасні технології програмування та апаратного забезпечення, що дозволяє споживачам швидко та безпечно купувати воду. У цьому розділі розглянемо основні методи та принципи, що використовуються в сучасних автоматах з продажу води, а також особливості їх функціонування.

#### Принципи роботи автоматів з продажу води

Сучасні автомати для продажу води використовують кілька ключових технологій:

Електронні платіжні системи. Більшість автоматів оснащені системами для прийому різних видів платежів, включаючи монети, банкноти, банківські картки та безконтактні платежі (наприклад, через NFC або мобільні додатки). Така різноманітність платіжних методів робить процес купівлі води швидким і зручним для користувачів.

Дозуючі системи. Один з основних елементів автомату – це система дозування, яка повинна точно контролювати об'єм виданої рідини. Сучасні технології використовують високоточні сенсори для вимірювання рівня рідини, що дозволяє уникнути перевитрати продукту та гарантувати користувачеві отримання необхідної кількості води.

Інтерактивний інтерфейс користувача. Новітні моделі автоматів обладнані сенсорними екранами або простими інтерфейсами, що дозволяють легко вибрати бажаний продукт і здійснити оплату. Це значно підвищує зручність користування.

## Типи автоматів для продажу води

Автомати з продажу води можна класифікувати за кількома критеріями:

Бутильовані автомати. Найпоширеніші пристрої для продажу води в пластикових пляшках різного об'єму. Ці автомати обладнані системами для контролю кількості товару та забезпечують автоматичну заміну порожніх слотів після продажу.

Стаціонарні автомати з розливом. Вони встановлюються у великих громадських місцях і забезпечують продаж води «на розлив». Користувач може прийти з власною тарою і наповнити її. Такі пристрої часто використовують у супермаркетах та на вулицях великих міст.

Автомати з багатофункціональними можливостями. Останнім часом все більше розробляються автомати, які можуть продавати не лише воду, а й інші напої (газовану воду, соки тощо). Такі пристрої оснащені більш складними системами дозування та контролю продуктів.

## Програмне забезпечення автоматів

Програмне забезпечення (ПЗ) є невід'ємною частиною роботи автоматів для продажу води. Основна функція ПЗ полягає в управлінні всіма процесами в автоматі: від обробки платежів до дозування води та видачі чека.

У сучасних автоматах використовуються такі технології:

Програмування контролерів. Контролери відповідають за управління всіма механічними і електронними компонентами автомата. Вони використовують прошивку на основі мікроконтролерів (наприклад, Arduino або Raspberry Pi), яка відповідає за комунікацію з сенсорами, платіжними системами та механізмами дозування.

Інтернет-рішення та віддалений моніторинг. Багато сучасних автоматів підтримують підключення до Інтернету для віддаленого моніторингу стану автомата, запасів води та інших параметрів. Це дозволяє операторам своєчасно реагувати на будь-які проблеми і оперативно поповнювати запас товару.

Обробка даних і аналітика. Деякі автомати обладнані системами збору даних, які аналізують продажі, споживчі звички та інші показники. Це допомагає операторам оптимізувати роботу автоматів і прогнозувати попит.

## 1.2 Аналіз програмних засобів управління автоматами

Програмне забезпечення є критичним елементом роботи автоматів для продажу рідин, оскільки воно забезпечує функціонування всієї системи — від взаємодії з користувачем до контролю процесів видачі товару. У цьому розділі розглянемо основні компоненти програмних засобів управління автоматами, їх архітектуру та особливості розробки на прикладі автоматів для продажу рідин.

### Архітектура програмних засобів для автоматів

Програмне забезпечення автоматів для продажу рідин зазвичай має багаторівневу архітектуру, що складається з кількох ключових модулів:

Нижній рівень (контролери). Цей рівень відповідає за управління апаратними компонентами автомата, такими як сенсори, насоси, дисплеї та платіжні системи. Програмне забезпечення для цього рівня зазвичай реалізується на базі вбудованих систем і мікроконтролерів, таких як Arduino або STM32, і відповідає за реальне керування механізмами.

Середній рівень (логіка автомата). На цьому рівні програмне забезпечення обробляє дії користувачів, координує платіжні транзакції та

керує дозуванням рідин. Основна логіка контролює, який продукт вибирає користувач, перевіряє наявність рідини в резервуарах, здійснює транзакції та керує всіма внутрішніми процесами автомата.

Верхній рівень (інтерфейс користувача та системи моніторингу). Цей рівень включає графічні або текстові інтерфейси, що дозволяють користувачам взаємодіяти з автоматом. Також цей рівень може включати програмне забезпечення для віддаленого моніторингу, яке забезпечує контроль за станом автомата і можливість його керування на відстані через інтернет-з'єднання.

При розробці програмного забезпечення для автоматів можуть використовуватися різні платформи і мови програмування. Найчастіше це:

Мови програмування для мікроконтролерів. Найпоширеніші мови для розробки на базі контролерів — це C, C++ і Python. Вони забезпечують швидкодію і простоту інтеграції з апаратним забезпеченням автомата. Такі платформи як Arduino IDE або PlatformIO надають зручні інструменти для розробки вбудованих рішень.

Операційні системи реального часу (RTOS). Для забезпечення надійної роботи критично важливих функцій автомата можуть використовуватися RTOS, такі як FreeRTOS або Zephyr. Вони дозволяють розподіляти ресурси між різними завданнями автомата, забезпечуючи виконання ключових операцій в реальному часі.

Платформи для віддаленого моніторингу. Для організації віддаленого моніторингу автоматів все частіше використовуються хмарні сервіси та IoT-платформи (наприклад, AWS IoT, Microsoft Azure IoT Hub). Вони дозволяють підключати автомат до інтернету і відстежувати його роботу через веб-інтерфейси або мобільні додатки.

## РОЗДІЛ 2. Вимоги до програмного забезпечення

### 2.1 Архітектура системи автомату

Архітектура системи автомату з продажу рідин побудована на базі мікроконтролера, який керує всіма процесами: обробкою введення від користувача, управлінням електронними компонентами (клапани, насос, лічильник води), а також відображенням інформації на РК-дисплеї.

Основні компоненти архітектури:

Мікроконтролер – головний елемент системи, що відповідає за виконання всіх програмних і апаратних завдань. Він обробляє сигнали від кнопок, датчиків рівня води, купюроприймача і монетоприймача, а також управляє насосом і клапаном подачі води.

Дисплей (LCD) – використовується для відображення інформації про статус роботи автомату, інструкцій для користувача, суми внесених коштів та об'єму наливої води. Ініціалізація дисплея здійснюється через бібліотеку LiquidCrystal.

Кнопка наливу води – фізична кнопка, що дозволяє користувачеві запустити процес наливу води після внесення відповідної суми. Натискання кнопки перевіряється через функцію `digitalRead`, а контроль над процесом наливу відбувається через клапан подачі води.

Лічильник імпульсів води – підраховує обсяг наливої рідини за допомогою імпульсів, які надходять від сенсора води. Код реалізує логіку підрахунку імпульсів через змінні `counterwater` і `lastStateWater`, що дозволяє визначати кількість рідини, що подається за певний проміжок часу.

Монетоприймач і купюроприймач – ці пристрої дозволяють користувачеві вносити кошти для оплати за воду. Мікроконтролер

отримує сигнал від цих пристроїв і збільшує лічильник внесених коштів, що зберігаються у змінних `countmoney` і `Litrmoney`. Лічильник відстежує суми, внесені за допомогою монет (10 коп., 50 коп., 1 грн) і купюр.

Клапан подачі води і насос – основні виконавчі елементи системи. Клапан подачі води відкривається для наливу води при натисканні кнопки і достатній кількості коштів на рахунку. Насос відповідає за подачу води з бака, якщо рівень води знаходиться в допустимих межах, що перевіряється за допомогою датчиків рівня (`ellow`, `elhigh`).

Датчики рівня води – контролюють рівень рідини в резервуарі. Якщо вода досягає нижнього або верхнього рівня, насос включається або вимикається відповідно. Це забезпечує безперервний контроль над запасами рідини і запобігає переповненню або пустому бакам.

Програмна логіка – весь код системи базується на циклі `loop()`, який постійно опитує стани входів і виконує відповідні дії на основі цих сигналів. Це забезпечує постійний контроль за процесами наливу і прийому платежів.

Архітектура системи автомату з продажу рідин є досить гнучкою і може бути модифікована в залежності від вимог проекту. Всі компоненти працюють у тісній взаємодії для забезпечення надійної роботи автомату та максимально зручного інтерфейсу для користувача.

## 2.2 Опис функціональних модулів системи

Система автомату з продажу рідин складається з кількох функціональних модулів, кожен з яких виконує окремі завдання і забезпечує коректну роботу пристрою в цілому. Взаємодія цих модулів дозволяє автомату обробляти введення користувача, контролювати налив рідини та виконувати оплату. Нижче наведено опис основних функціональних модулів системи, заснований на розробленому коді.

## Модуль введення даних користувача

Цей модуль відповідає за отримання введення від користувача, зокрема натискання кнопок та прийом оплати через монетоприймач і купюроприймач. Функціонал включає:

Кнопка наливу води: Контролюється через порт введення/виведення мікроконтролера. Функція `digitalRead` перевіряє стан кнопки і дозволяє почати процес наливу після перевірки умов (наявність достатньої кількості коштів та об'єму води в резервуарі).

Монетоприймач і купюроприймач: Прийом оплати реалізується через змінні `countmoney` та `Litrmoney`, які накопичують суми, внесені користувачем. Для кожного типу монет чи купюр є окремий сигнал, що передається до мікроконтролера та додається до загальної суми внесених коштів.

## Модуль управління подачею води

Модуль відповідає за керування процесом наливу води відповідно до введення користувача та умов, визначених у коді. Його функції включають:

Клапан подачі води: Контролюється мікроконтролером, який відкриває клапан для подачі води за умови, що користувач вніс достатньо коштів і натиснув кнопку наливу. Використовується функція `digitalWrite` для управління станом клапана.

Насос: Вмикається та вимикається залежно від рівня води в резервуарі. Насос відповідає за подачу води з основного резервуару в клапан, якщо датчики рівня води (`elhigh`, `ellow`) сигналізують про допустимий рівень рідини.

## Модуль підрахунку об'єму наливої рідини

Цей модуль реалізує контроль за кількістю води, що подається користувачу. Він використовує імпульсний лічильник, який дозволяє точно визначати об'єм наливої рідини. Функціонал включає:

Лічильник імпульсів води: Кожен імпульс, який генерується сенсором води, обробляється через змінні `counterwater` та `lastStateWater`. Кількість імпульсів визначає кількість налитих літрів води. Як тільки користувач отримує необхідну кількість рідини, клапан закривається.

#### Модуль відображення інформації

Модуль відповідає за взаємодію з користувачем шляхом відображення актуальної інформації на дисплеї. Це включає:

LCD-дисплей: Виводить повідомлення про статус автомату, інструкції для користувача, інформацію про внесену суму та обсяг наливої води. За допомогою бібліотеки `LiquidCrystal` ініціалізується дисплей і відображається відповідна інформація, що оновлюється в режимі реального часу.

#### Модуль контролю рівня води

Важливою частиною системи є контроль рівня води у резервуарі, що забезпечується через модуль датчиків рівня. Його функції включають:

Датчики верхнього і нижнього рівня: Датчики контролюють рівень води в резервуарі. Якщо рівень води опускається нижче допустимого, система сигналізує про необхідність поповнення резервуару. Якщо рівень досягає верхньої межі, подача води припиняється для уникнення переповнення.

#### Модуль керування оплатою

Модуль забезпечує облік внесених користувачем коштів та перевірку, чи достатньо суми для початку наливу води. Він реалізований



через наступні елементи:

Монетоприймач і купюроприймач: Як тільки користувач вносить монети або купюри, сума автоматично накопичується. Код перевіряє внесену суму і порівнює її з необхідною для покупки води (змінні countmoney та Litrmoney).

Програмний модуль логіки управління

Весь цикл функціонування автомату побудований навколо основної програми (loop()), що постійно опитує стани всіх входів і здійснює управління виконавчими механізмами. У цьому модулі реалізовано:

Обробка введення від користувача (кнопки та оплата).

Контроль подачі води на основі об'єму внесених коштів.

Відображення інформації на дисплеї.

Підрахунок імпульсів та об'єму води.

Автоматичне вимкнення або увімкнення насосу в залежності від рівня води.

Завдяки модульній структурі система автомату легко піддається розширенню та налаштуванню. Кожен з модулів може бути змінений або доповнений для покращення функціональних можливостей пристрою, а також адаптований під різні типи рідин або вимоги до автомату.

### 2.3 Вибір апаратного забезпечення

Проектування автомату для продажу рідин вимагає ретельного підбору апаратного забезпечення, що забезпечить надійну роботу системи, точне вимірювання об'єму рідини, контроль оплати та зручну взаємодію з користувачем. Вибір апаратних компонентів здійснювався з урахуванням потреб у продуктивності, точності та економічності, що

відповідає вимогам автоматів для роздачі води. У цьому розділі розглянуто основні апаратні компоненти, використані у проекті, та їх функціональне призначення.

## 1. Мікроконтролер

Центральним елементом апаратної частини системи є мікроконтролер, який забезпечує управління всіма компонентами автомату. Для цього проекту був обраний Arduino Uno через його доступність, простоту у використанні та достатню кількість цифрових і аналогових входів/виходів для підключення периферійних пристроїв. Основні причини вибору Arduino Uno:

14 цифрових входів/виходів (6 з яких можуть використовуватися для ШІМ-сигналів), що достатньо для підключення таких компонентів, як кнопки, датчики рівня, монетоприймачі та насоси.

6 аналогових входів, що дозволяє підключати сенсори рівня рідини та інші аналогові компоненти.

Наявність достатньої пам'яті (32 КВ флеш-пам'яті) для зберігання програми керування всіма елементами системи.

## 2. Дисплей (LCD)

Для відображення інформації користувачу використовується LCD-дисплей 16x2. Він дозволяє виводити текстові повідомлення, такі як інструкції для користувача, обсяг наливої рідини, стан внесених коштів тощо. Вибір цього дисплея був зумовлений його низьким енергоспоживанням, простотою в інтеграції з Arduino через бібліотеку LiquidCrystal і достатньою кількістю символів для виведення базової інформації.

### 3. Монетоприймач і купюроприймач

Для забезпечення функції прийому оплати використовуються монетоприймач і купюроприймач, які підключені до мікроконтролера. Ці пристрої генерують сигнали після внесення монет або купюр, що дозволяє системі визначати кількість отриманих коштів. Причини вибору даних пристроїв:

Простота інтеграції з Arduino через цифрові входи.

Швидка реакція на внесення коштів та точний підрахунок сум.

Підтримка різних номіналів монет і купюр, що забезпечує гнучкість системи.

### 4. Клапан для подачі води

Для регулювання потоку води використовується електромагнітний клапан, який відкривається або закривається за командою від мікроконтролера через цифрові виходи. Основні характеристики, що зумовили вибір цього компонента:

Надійність в умовах роботи з рідинами.

Можливість точного контролю за подачею рідини залежно від внесених коштів.

Легка інтеграція з реле для керування через мікроконтролер.

### 5. Насос

Для перекачування води з резервуару використовується водяний насос, який підключається через реле до мікроконтролера. Насос включається тільки за умови достатнього рівня води в резервуарі, що контролюється датчиками. Основні характеристики:

Підтримка безперервного потоку води для забезпечення

рівномірного наливу.

Працює під керуванням мікроконтролера через реле, що забезпечує безпеку та надійність роботи.

#### 6. Датчики рівня води

Для контролю рівня рідини в резервуарі використовується два датчики рівня – верхнього і нижнього рівнів. Вони забезпечують:

Автоматичне вимкнення насоса при низькому рівні рідини.

Захист від переливання за допомогою верхнього датчика рівня. Ці датчики підключені до цифрових входів Arduino, що дозволяє швидко реагувати на зміни рівня води.

#### 7. Лічильник імпульсів для контролю наливу

Для точного вимірювання об'єму рідини, яка подається користувачу, використовується лічильник імпульсів. Цей датчик генерує імпульси залежно від кількості води, що проходить через систему. Переваги цього підходу:

Висока точність вимірювання об'єму.

Можливість швидкого налаштування кількості води, що наливатиметься на один імпульс.

Інтеграція з мікроконтролером для автоматичного керування процесом наливу.

#### 8. Кнопка керування наливом

Для початку процесу наливу користувачеві надається кнопка, підключена до одного з цифрових входів Arduino. При натисканні кнопки мікроконтролер перевіряє умови (внесення коштів, рівень води) і запускає налив рідини. Вибір цього компонента зумовлений:

Простотою інтеграції.

Надійністю в роботі при частому використанні.

## 9. Реле

Для управління насосом і клапаном використовується реле, яке служить проміжною ланкою між низьковольтними сигналами мікроконтролера та високовольтними компонентами системи. Реле забезпечує безпечне керування потоком води і захищає електронні компоненти від перевантажень.

Усі апаратні компоненти системи автомату з продажу рідин були обрані з урахуванням їх сумісності з мікроконтролером Arduino Uno, а також здатності забезпечувати необхідний рівень функціональності, надійності та точності. Важливою частиною проекту було забезпечення простоти інтеграції компонентів та можливості їх подальшої модернізації або заміни для покращення продуктивності автомату.

### 2.4 Алгоритми роботи автомату

Алгоритми роботи автомату з продажу рідин є основою програмного забезпечення, що контролює всі етапи процесу взаємодії між користувачем і системою. Основні завдання алгоритмів включають обробку платежів, керування наливом рідини, контроль рівня води, а також відображення інформації на дисплеї. У цьому розділі буде детально розглянуто основні алгоритми, що забезпечують функціонування автомату.

#### Алгоритм очікування введення

Цей алгоритм є початковим станом автомату, коли система знаходиться в режимі очікування взаємодії з користувачем. На цьому етапі система відображає на дисплеї вітальне повідомлення та

інформацію про вартість рідини. У режимі очікування автомат постійно перевіряє такі події:

Внесення монет чи купюр.

Натискання кнопки для початку наливу.

Алгоритм роботи: вивести на дисплей повідомлення "Внесіть монети". Постійно опитувати монетоприймач і купюроприймач для отримання сигналу про внесення коштів. Як тільки отримані кошти, перейти до наступного етапу.

Алгоритм обробки оплати

Після отримання сигналу від монетоприймача або купюроприймача система запускає алгоритм обробки оплати. Він відповідає за підрахунок загальної суми внесених коштів і перевірку їхньої достатності для здійснення покупки. Якщо внесено недостатньо коштів, система продовжує очікувати додаткового внесення.

Алгоритм роботи: після отримання сигналу від монетоприймача, додати відповідний номінал до загальної суми. Вивести на дисплей поточну суму внесених коштів. Якщо сума менша за вартість рідини, відобразити на дисплеї повідомлення про необхідність додаткових коштів і продовжувати очікування. Якщо сума достатня, перейти до наступного етапу – початку наливу.

Алгоритм контролю наливу

Алгоритм контролю наливу забезпечує точне дозування рідини відповідно до внесеної суми. Після завершення процесу оплати система починає налив, керуючи насосом і клапаном. Цей алгоритм включає кілька ключових моментів: перевірка рівня води в резервуарі перед початком наливу. Відкриття клапана і запуск насоса. Контроль за об'ємом

рідини, що наливається, за допомогою лічильника імпульсів.

Алгоритм роботи: перевірити рівень води за допомогою датчиків. Якщо рівень недостатній, вивести на дисплей повідомлення про недоступність рідини і завершити процес. Відкрити клапан і запустити насос. Підраховувати імпульси з лічильника для визначення кількості наливої рідини. Після досягнення необхідної кількості рідини вимкнути насос і закрити клапан. Вивести на дисплей повідомлення про завершення наливу.

#### Алгоритм контролю рівня рідини

Цей алгоритм постійно перевіряє рівень рідини в резервуарі за допомогою верхнього і нижнього датчиків рівня води. Його основна мета – захист системи від роботи насоса на суху (коли рівень води занадто низький) або переповнення резервуара. В разі недостатнього рівня рідини, система припиняє всі операції та виводить відповідне повідомлення.

Алгоритм роботи: постійно опитувати датчики рівня води. Якщо рівень нижчий за допустимий, заблокувати функцію наливу і вивести повідомлення "Недостатньо води". Якщо рівень нормальний, дозволити запуск наливу.

#### Алгоритм обробки кнопок

Усі взаємодії користувача з автоматом через кнопки (початок наливу, скасування операції тощо) обробляються окремим алгоритмом. Цей алгоритм опитує стан кнопок і запускає відповідні дії.

Алгоритм роботи: опитувати стан кнопки початку наливу. Якщо кнопка натиснута, перевірити умови (наявність коштів, рівень води). Якщо всі умови виконані, почати процес наливу.

## Алгоритм інформування користувача

Протягом всього процесу взаємодії з автоматом користувачеві надається актуальна інформація через дисплей. Цей алгоритм відповідає за відображення повідомлень на всіх етапах – від очікування оплати до завершення наливу.

Алгоритм роботи: виводити на дисплей поточний стан системи (очікування оплати, процес наливу, видача решти). Відображати інформацію про поточну суму внесених коштів і об'єм наливої рідини.

Алгоритми, що забезпечують роботу автомату з продажу рідин, охоплюють усі аспекти функціонування системи: від обробки платежів і наливу до контролю рівня рідини та інформування користувача. Усі алгоритми розроблені з урахуванням потреб у надійності, точності та зручності для кінцевого користувача.



## РОЗДІЛ 3. Опис алгоритмів реалізації вибору продукту

### 3.1 Розрахунок вартості та контроль оплати

Автомат з продажу рідин забезпечує процес прийому платежів та контролю за правильністю оплати для надання користувачеві відповідної кількості рідини. Цей процес включає розрахунок вартості рідини, контроль внесених коштів і повернення решти у разі надлишкової оплати. Функціонування цієї системи засноване на використанні апаратних і програмних компонентів, таких як монетоприймач, купюроприймач та блок контролю суми.

#### Розрахунок вартості рідини

Вартість рідини, яку користувач планує придбати, залежить від об'єму наливу та встановленої ціни за літр. Основними параметрами для розрахунку є:

Ціна за 1 літр рідини.

Вибраний користувачем об'єм рідини (за замовчуванням, фіксований об'єм або на основі оплати).

Розрахунок виконується за простою формулою:

Вартість=Об'єм×Ціна за літр

Наприклад, якщо користувач хоче придбати 5 літрів води, а ціна за літр становить 1 грн, то загальна вартість буде:

$5\text{л} \times 1\text{ грн/л} = 5\text{ грн}$

Вартість відображається на дисплеї автомату до початку прийому оплати.

#### Процес прийому оплати

Система прийому оплати побудована на основі монетоприймача та купюроприймача, які фіксують номінали внесених користувачем грошей. Після внесення кожної монети або купюри система автоматично оновлює поточну суму внесених коштів та відображає її на дисплеї.

Алгоритм прийому оплати включає такі основні етапи:

Користувач вносить монети або купюри у відповідні приймачі.

Автомат обчислює загальну суму внесених коштів.

Якщо внесена сума менша за вартість замовленого об'єму рідини, система продовжує очікувати додаткові кошти.

Якщо сума дорівнює або перевищує необхідну вартість, система переходить до етапу наливу рідини або видачі решти.

Контроль достатності оплати

Після кожної операції внесення коштів система виконує перевірку, чи внесено достатню кількість грошей для оплати бажаного об'єму рідини. Це здійснюється через порівняння загальної суми внесених грошей із вартістю обраного об'єму рідини.

Алгоритм контролю оплати:

Якщо сума внесених коштів менша за необхідну вартість, система відображає на дисплеї повідомлення про недостатню кількість грошей.

Користувач може продовжити вносити додаткові кошти, доки загальна сума не досягне або не перевищить вартість рідини.

Якщо сума достатня, система автоматично переходить до наступного етапу.

Обробка помилок під час оплати

Під час процесу прийому та обробки оплати можуть виникати різні помилки, такі як неправильне розпізнавання монет або збій у роботі купюроприймача. Для цього система містить алгоритми обробки помилок:

Якщо монета не розпізнана або вона несправна, система повертає її користувачеві.

У випадку збою купюроприймача система зупиняє процес оплати і відображає повідомлення про помилку на дисплеї.

Процес розрахунку вартості та контролю оплати є одним із ключових елементів роботи автомату з продажу рідин. Він забезпечує не тільки прийом коштів, але й гарантує коректність розрахунків, точне дозування рідини відповідно до внесених коштів. Програмні алгоритми разом з апаратними компонентами, такими як монетоприймач та купюроприймач, забезпечують надійність і точність цього процесу.

### 3.2 Алгоритми дозування та видачі продукту

У автоматах з продажу рідин важливу роль відіграє точність і надійність процесів дозування та видачі продукту. Ці операції забезпечують відповідність об'єму рідини, що відпускається користувачеві, сумі, яку він сплатив, або обсягу, заданому оператором автомату. Алгоритми дозування та видачі рідини опираються на точне управління клапанами, сенсорами рівня та лічильниками об'єму. У даному розділі розглянуто принципи побудови алгоритмів дозування, основні етапи процесу видачі рідини та контроль виконання цих операцій.

#### Основні принципи дозування

Дозування рідини в автоматі зазвичай здійснюється шляхом контролю клапанів, що регулюють потік рідини з резервуару до

видаткового крана. Контроль точності дозування базується на наступних принципах:

Використання лічильника об'єму для відстеження кількості рідини, яка протікає через систему.

Налаштування системи таким чином, щоб вона відкривала і закривала клапан в залежності від досягнення необхідного об'єму рідини.

Наприклад, якщо користувач обирає 1 літр води, система має відкрити клапан і слідкувати за кількістю рідини, що пройшла через витратомір, до досягнення вказаного об'єму, після чого закриває клапан.

Етапи процесу дозування

Алгоритм дозування рідини можна розділити на кілька основних етапів:

Ініціація процесу дозування: Після внесення коштів або вибору об'єму система розпочинає процес наливу. Клапан відкривається для початку потоку рідини.

Контроль об'єму рідини: Під час наливу лічильник об'єму відслідковує кількість рідини, що проходить через систему. Система порівнює поточний об'єм з вибраним значенням.

Зупинка наливу: Як тільки об'єм рідини досягає заданого значення, система закриває клапан, припиняючи потік.

Очищення лінії (за необхідності): Для запобігання затримки рідини в трубах, система може виконувати очистку труб шляхом додаткового видування залишків рідини після закриття клапану.

Контроль видачі рідини

Процес видачі рідини має бути контрольованим для забезпечення точності та безпеки операції. Основними компонентами, які забезпечують цей контроль, є:

Витратоміри або сенсори рівня: Вони дозволяють точно визначати кількість рідини, що була видана, і сигналізують про досягнення необхідного об'єму.

Клапани та насоси: Управління клапанами, які відкривають і закривають подачу рідини, виконується мікроконтролером на основі показників сенсорів.

Запобіжні механізми: У разі виникнення несправностей (наприклад, перевищення часу видачі або збої у витратомірах) система повинна автоматично зупиняти процес наливу і повідомляти користувача про помилку.

#### Алгоритм дозування та видачі рідини

Типовий алгоритм дозування рідини в автоматах виглядає наступним чином:

1. Отримати від користувача запит на об'єм рідини або розрахувати об'єм на основі внесеної суми.
2. Перевірити наявність достатньої кількості рідини в резервуарі.
3. Відкрити клапан для початку наливу.
4. Відстежувати кількість виданої рідини за допомогою витратоміра або лічильника об'єму.
  - 4.1 Якщо об'єм рідини досягне заданого значення – перейти до п. 5.
  - 4.2 Якщо виникає помилка (наприклад, несправність витратоміра)

– зупинити процес і повідомити про помилку.

5. Закрити клапан після досягнення необхідного об'єму.

6. Виконати очистку системи (якщо необхідно).

7. Завершити операцію, повідомивши користувача про успішне завершення наливу.

Кінець.

Можливі помилки під час наливу

Під час виконання операцій дозування та видачі рідини можуть виникати різні помилки або збої. Для таких випадків алгоритм передбачає відповідні дії:

Перевищення часу наливу: Якщо час наливу перевищує допустимий ліміт, система зупиняє процес і закриває клапан.

Несправність сенсорів або витратомірів: У випадку, коли сенсори не передають коректні дані або не можуть визначити об'єм рідини, процес наливу припиняється, і користувачу надається повідомлення про помилку.

Нестача рідини в резервуарі: Якщо виявлено недостатню кількість рідини, система не починає налив і повідомляє про це користувача.

Оптимізація процесу дозування

Для підвищення ефективності і точності процесу дозування можуть бути використані такі методи оптимізації:

Калібрування витратомірів: Регулярна перевірка і калібрування датчиків дозволяє уникнути похибок у вимірюванні об'єму рідини.

Динамічний контроль швидкості потоку: Впровадження

алгоритмів, що дозволяють регулювати швидкість подачі рідини в залежності від залишкового об'єму (наприклад, зменшення швидкості при наближенні до кінцевого об'єму), дозволяє уникати перевищення об'єму наливу.

Попередження користувача: Інформаційне сповіщення користувача про процес наливу (наприклад, "Налив завершено" або "Залишилося Х мл") підвищує зручність і надійність використання автомату.

Алгоритми дозування та видачі продукту є критичними елементами роботи автомату з продажу рідин. Вони забезпечують точне вимірювання виданої рідини, контроль об'єму та надійну роботу обладнання. За допомогою точних сенсорів, клапанів і програмних алгоритмів, автомат може гарантувати користувачам безпомилкову видачу продукту відповідно до сплачених коштів.

### 3.3 Управління сенсорами та акторами

В автоматах з продажу рідин ключову роль у функціонуванні системи відіграють сенсори та актори (пристрої, що виконують фізичні дії на основі сигналів від контролера). Сенсори забезпечують збір даних про стан системи, таких як рівень рідини, тиск, температура, або об'єм виданого продукту. Актори, в свою чергу, виконують фізичні дії, наприклад, відкривають клапани, запускають насоси або контролюють інші механізми, що відповідають за видачу рідини. Коректне програмування взаємодії між цими компонентами є основою надійної та точної роботи автомату.

#### Сенсори в автоматах з продажу рідин

У системах продажу рідин можуть використовуватися різні види сенсорів для забезпечення точного контролю за процесами наливу і

продажу. До них належать:

Витратоміри — відслідковують кількість рідини, що пройшла через трубопровід, та передають дані до контролера для визначення об'єму наливої рідини.

Сенсори рівня — визначають кількість рідини у резервуарах. Вони можуть бути поплавковими, ємнісними або ультразвуковими, залежно від конструкції автомату.

Температурні сенсори — контролюють температуру рідини для забезпечення належних умов зберігання та наливу, особливо для автоматів, що реалізують охолоджені або підігріті напої.

Тиск-сенсори — вимірюють тиск у трубопроводах для моніторингу системи і забезпечення рівномірного потоку рідини.

#### Актори в системі

Актори є виконавчими елементами системи і відіграють вирішальну роль у реалізації команд, що надходять від контролера. До основних акторів у системі продажу рідин належать:

Електромагнітні клапани — відповідають за відкриття та закриття шляхів подачі рідини. Коли система отримує сигнал про початок наливу, клапан відкривається, а після досягнення потрібного об'єму рідини — закривається.

Насоси — використовуються для подачі рідини з резервуарів під тиском. Насоси можуть бути необхідні в системах з високою в'язкістю рідини або у випадках, коли рідина повинна подаватися на велику висоту чи відстань.

Світлодіоди або дисплеї — забезпечують зворотній зв'язок користувачеві, показуючи стан системи (наприклад, готовність до наливу,



помилки або завершення операції).

Звукові сигнали — можуть використовуватися для повідомлення про успішну транзакцію, початок або завершення видачі рідини.

### 3.4 Програмування та тестування системи

Програмування автоматів з продажу рідин є ключовим етапом розробки, що передбачає створення програмної логіки для керування всіма компонентами системи: сенсорами, акторами, дисплеями та інтерфейсами користувача. Мета цього процесу — забезпечити коректну взаємодію між апаратними елементами, гарантувати правильність виконання бізнес-логіки (зокрема, обробка оплат і видача товару) та забезпечити надійність роботи пристрою.

#### Програмування системи

Програмування автомату з продажу рідин включає розробку програмного забезпечення, яке управляє всіма аспектами роботи автомату. Основні етапи програмування включають:

Ініціалізація системи: Під час запуску системи відбувається ініціалізація всіх підключених пристроїв — сенсорів, акторів та інтерфейсів. Контролер отримує початкові дані від сенсорів (рівень рідини, тиск тощо) і переводить систему в готовий до роботи стан.

Контроль сенсорів: Код забезпечує постійний моніторинг сенсорів для отримання даних про поточний стан системи. Наприклад, сенсори рівня контролюють кількість рідини в резервуарі, а витратоміри стежать за обсягом рідини, яка видається користувачеві.

Управління акторами: Програмна логіка відповідає за контроль виконавчих механізмів. Наприклад, після успішної оплати відбувається відкриття клапану, щоб налити певний обсяг рідини, після чого клапан

закривається.

Взаємодія з користувачем: Користувач взаємодіє з автоматом через інтерфейс (кнопки, сенсорний екран). Програмне забезпечення обробляє введення даних, дозволяючи користувачеві вибрати тип рідини, її об'єм та здійснити оплату.

Обробка транзакцій: Програмний модуль для обробки транзакцій контролює процес оплати, верифікує надходження коштів та передає сигнал для видачі продукту.

### Тестування системи

Тестування є необхідним етапом для виявлення помилок у програмній логіці та забезпечення коректної роботи всіх компонентів системи. Основні етапи тестування включають:

Тестування функціональних модулів: Кожен модуль програмного забезпечення тестується окремо для перевірки його функціональності. Наприклад, тестування роботи сенсорів, правильності управління клапанами та іншими акторами.

Тестування бізнес-логіки: Важливо перевірити, чи правильно обробляються сценарії користувацької взаємодії, зокрема вибір рідини, об'єм наливу, коректність відображення вартості та обробка платежів.

Інтеграційне тестування: Після перевірки окремих модулів виконується інтеграційне тестування, яке перевіряє, як усі модулі працюють разом. Наприклад, система повинна коректно взаємодіяти з платіжними терміналами, забезпечувати видачу рідини після підтвердження оплати та вести облік виданих обсягів.

Навантажувальне тестування: Це тестування допомагає виявити, як

система працює під час інтенсивного використання. Важливо, щоб система залишалася стабільною при великій кількості запитів та транзакцій.

#### Використання симуляторів для тестування

У процесі розробки системи можна використовувати симулятори для перевірки програмної логіки до підключення реального обладнання. Це дозволяє уникнути пошкодження апаратних компонентів та зекономити час. За допомогою симуляцій можна перевірити, як програма реагує на різні сценарії роботи сенсорів, як вона обробляє помилки або екстрені ситуації.

Програмування та тестування автоматів з продажу рідин є невід'ємною частиною процесу розробки, яка дозволяє забезпечити високу надійність системи та задовольнити потреби користувачів.

## РОЗДІЛ 4. Вибір мови програмування та інструментів розробки

### 4.1 Основні етапи програмування системи

Програмування автоматів з продажу рідин є складним процесом, що включає розробку програмної логіки для забезпечення ефективної взаємодії між різними компонентами системи: сенсорами, акторами, інтерфейсами користувача та модулями контролю транзакцій. Основними етапами програмування такої системи є:

#### 1. Аналіз вимог та проектування системи

На першому етапі необхідно визначити ключові функції автомату, такі як продаж рідин, інтеграція з платіжними системами, моніторинг рівня рідини та керування механізмами видачі продукту. Це включає аналіз вимог до системи, зокрема, як вона має взаємодіяти з користувачем, якими даними управляти та як реагувати на зовнішні події (наприклад, завершення наливу чи завершення транзакції).

Опис функціональних вимог: визначаються основні функції, такі як обробка оплат, дозування рідини, видача продукту, взаємодія з сенсорами та керування акторними пристроями.

Проектування архітектури: визначаються основні модулі системи та їх взаємодія між собою. Зазвичай система включає модуль керування сенсорами, керування акторними пристроями, обробки оплат та інтерфейсу користувача.

#### 2. Розробка програмних модулів

На цьому етапі відбувається безпосереднє написання коду, що реалізує необхідні функції системи. Програмне забезпечення для автомату розробляється модульно, що полегшує його тестування та подальше вдосконалення.

Модуль управління сенсорами: цей модуль обробляє дані, що надходять від різних сенсорів (витратомірів, сенсорів рівня, температури). Його функція полягає у зборі інформації про стан системи, яку контролер використовує для прийняття рішень щодо видачі рідини.

Модуль керування акторними пристроями: відповідає за роботу механізмів, таких як насоси, клапани та дозатори. Програмне забезпечення керує цими пристроями на основі отриманих сигналів від сенсорів та команд від основного контролера.

Модуль обробки транзакцій: виконує взаємодію з платіжною системою, перевіряє валідність транзакції та передає сигнал для запуску процесу наливу рідини після успішної оплати.

Модуль інтерфейсу користувача: відповідає за відображення інформації для користувача та обробку його запитів. Це може включати вибір об'єму рідини, відображення вартості, підтвердження транзакції або повідомлення про завершення наливу.

### 3. Інтеграція модулів

Після розробки окремих модулів важливим етапом є інтеграція всіх компонентів у єдину систему. Цей процес полягає у тому, щоб забезпечити правильну взаємодію між сенсорами, акторами, платіжною системою та інтерфейсом користувача. Всі модулі повинні обмінюватися даними у реальному часі, щоб забезпечити плавну та ефективну роботу автомату.

### 4. Тестування системи

Тестування є важливим етапом програмування, яке дозволяє виявити та усунути можливі помилки або недоліки в роботі системи. Основними типами тестування є:

Функціональне тестування: перевіряються окремі модулі системи на відповідність вимогам. Наприклад, чи коректно спрацьовують сенсори рівня, чи правильно керуються клапани.

Інтеграційне тестування: перевіряється взаємодія між різними модулями. Наприклад, після оплати система повинна автоматично запустити дозування і контролювати процес наливу через відповідні сенсори.

Тестування надійності: перевіряється стійкість системи до навантажень та її здатність працювати без збоїв у режимі багатократного використання.

## 5. Налаштування та оптимізація

Після тестування система налаштовується для роботи на реальному обладнанні. Це включає:

Калібрування сенсорів: для точного визначення рівня рідини або кількості наливої рідини, кожен сенсор повинен бути правильно налаштований.

Оптимізація логіки: перевіряється, чи всі процеси виконуються з мінімальними затримками, і чи відповідає система вимогам до швидкості обробки запитів.

## 6. Впровадження та підтримка

Після завершення тестування і налаштування система готова до впровадження. Важливо забезпечити можливість подальшої підтримки та оновлення програмного забезпечення для адаптації до нових вимог або вдосконалення функцій автомату.

Таким чином, основні етапи програмування системи автомату з продажу рідин включають аналіз вимог, розробку модулів, інтеграцію,

тестування, налаштування і подальше впровадження системи.

## 4.2 Тестування програмного забезпечення та моделювання роботи автомата

Тестування програмного забезпечення та моделювання роботи автомату з продажу рідин є критично важливим етапом для забезпечення надійної та безперебійної роботи системи. Ці процеси дозволяють виявити помилки на ранніх стадіях, перевірити коректність алгоритмів та забезпечити стабільність системи в реальних умовах експлуатації.

### 1. Методологія тестування

У процесі розробки програмного забезпечення для автоматів з продажу рідин було використано кілька підходів до тестування, зокрема:

Модульне тестування. Кожен функціональний модуль системи (платіжні системи, дозування, робота сенсорів) тестується окремо для забезпечення його правильної роботи. Це дозволяє виявити помилки у конкретних функціях і швидко їх виправити.

Інтеграційне тестування. Після успішного модульного тестування окремі компоненти об'єднуються для перевірки їх взаємодії. Це дозволяє переконатися, що різні частини системи працюють гармонійно та без конфліктів.

Системне тестування. На етапі системного тестування проводиться перевірка всієї системи в цілому. Це включає симуляцію повного робочого процесу автомата: від прийому замовлення до видачі продукту та завершення транзакції. Основна мета цього тестування – забезпечити правильну роботу системи в реальних умовах.

### 2. Тестування сценаріїв використання

Тестування сценаріїв використання полягає у перевірці різних

варіантів взаємодії користувача з автоматом:

Тестування платіжних сценаріїв. Перевіряються всі можливі варіанти оплати: успішна оплата, відхилення транзакції, недостатньо коштів на рахунку, скасування транзакції користувачем.

Тестування видачі продукту. Моделюються сценарії, де користувач вводить різні обсяги рідини, щоб переконатися у правильному дозуванні та видачі продукту.

Тестування взаємодії з сенсорами. Проводяться тести роботи сенсорів рівня рідини, датчиків температури та інших сенсорних систем для впевненості у правильності їх реакції на зовнішні умови.

### 3. Моделювання роботи автомата

Для зменшення ризиків і підвищення ефективності тестування, особливо перед запуском системи в реальну експлуатацію, використовується програмне моделювання роботи автомата. Моделювання дозволяє протестувати систему в умовах, максимально наближених до реальних, але без необхідності використання фізичного обладнання.

Програмні симуляції. Симуляції відтворюють взаємодію з усіма сенсорами та акторними пристроями (мотор дозування, насоси, реле), використовуючи програмну модель для контролю їхньої роботи.

Навчальні сценарії. Проводяться навчальні сценарії для операторів, де відтворюються можливі проблеми або збої (наприклад, помилкова відмова сенсора), щоб навчити операторів швидко реагувати на подібні ситуації.

Навантажувальне тестування. Використовується для перевірки



того, як система буде працювати при великій кількості одночасних транзакцій або тривалій безперервній роботі. Це допомагає виявити можливі точки збоїв при високих навантаженнях.

#### 4. Аналіз результатів тестування

На завершальному етапі тестування проводиться детальний аналіз отриманих результатів. Якщо виявлені помилки або недоліки в роботі системи, розробники проводять їх виправлення. Повторне тестування необхідне для підтвердження, що виправлення не вплинули на інші частини системи.

Таким чином, процес тестування та моделювання роботи автомату є одним з ключових етапів розробки, що дозволяє забезпечити стабільну та безпечну роботу системи в реальних умовах експлуатації.

## РОЗДІЛ 5. Оцінка вартості розробки та впровадження

### 5.1 Оцінка економічної ефективності системи

Оцінка економічної ефективності системи автоматів з продажу рідин є важливим етапом розробки та впровадження подібних технологій. Вона дозволяє визначити рівень витрат на створення та експлуатацію автомату, а також потенційний прибуток від його використання. Ефективність роботи системи можна оцінювати через аналіз ряду ключових факторів, серед яких початкові інвестиції, витрати на експлуатацію та обслуговування, окупність та прибутковість автомату.

#### 1. Початкові інвестиції

Для оцінки економічної ефективності необхідно спершу врахувати початкові витрати на створення автомату, які включають:

Витрати на апаратне забезпечення. Включають вартість датчиків, дозуючих механізмів (насосів, клапанів), дисплеїв, контролерів, модулів зв'язку, корпусу та інших механічних компонентів.

Витрати на програмне забезпечення. Окрім апаратної складової, необхідно розробити програмне забезпечення для контролю системи. Це може включати розробку ПЗ для управління датчиками, актору, оплати, а також для моніторингу та управління через віддалені сервіси.

Інсталяційні витрати. Включають витрати на встановлення автомату на місці експлуатації, підключення до мереж водопостачання та електрики, а також налаштування та калібрування системи перед введенням в експлуатацію.

#### 2. Експлуатаційні витрати

Після інсталяції автомату важливо оцінити експлуатаційні витрати, які включають:

Енергоспоживання. Електроспоживання насосів, датчиків та інших електричних компонентів автомату може впливати на загальні витрати. Оптимізація роботи компонентів, таких як використання енергоефективних двигунів і сенсорів, дозволить знизити енерговитрати.

Обслуговування та технічна підтримка. Регулярне технічне обслуговування автомату необхідне для підтримання його стабільної роботи, що включає очищення, калібрування та заміну компонентів. Вартість таких робіт залежить від складності та тривалості експлуатації автомату.

Запасні частини та ремонт. Протягом часу деякі компоненти автомату можуть зношуватися, що потребує їх заміни. Це особливо стосується насосів, клапанів та інших механічних частин, що контактують з рідиною.

### 3. Прибутковість та окупність

Прибутковість автомату з продажу рідин залежить від кількох факторів:

Ціна продукту. Вартість рідини, що продається через автомат, визначає рівень доходу. Для оптимальної рентабельності необхідно знайти баланс між ціною та собівартістю рідини, враховуючи конкуренцію на ринку та попит серед споживачів.

Попит та частота використання. Чим більше користувачів регулярно використовуватимуть автомат, тим швидше окупляться початкові інвестиції. Автомат може бути розташований у місцях з великим потоком людей (торгові центри, вокзали, парки), що підвищує його рентабельність.

Окупність системи. Важливо оцінити термін окупності автомату –

час, за який отриманий дохід повністю покриє початкові інвестиції та експлуатаційні витрати. Термін окупності залежить від інтенсивності використання автомату та його надійності, що зменшує витрати на ремонт і обслуговування.

#### 4. Оптимізація витрат та збільшення доходів

Задля підвищення економічної ефективності автомату можна впровадити такі рішення:

Оптимізація енерговитрат. Використання енергоефективних компонентів, як-от датчики з низьким енергоспоживанням та насоси з інтелектуальним керуванням, дозволяє знизити загальні витрати на електроенергію.

Автоматизація обслуговування. Впровадження віддаленого моніторингу стану автомату через IoT-системи дозволяє знизити витрати на технічне обслуговування та попереджувати поломки ще до їх виникнення.

Збільшення асортименту продуктів. Можливість реалізовувати різні типи рідин через один автомат дозволить залучити більшу кількість споживачів і збільшити дохід.

#### 5. Моделювання та прогнозування ефективності

Для точнішої оцінки економічної ефективності можна застосувати моделі прогнозування на основі зібраних даних про використання автомату:

Моделювання сценаріїв. Програмне забезпечення дозволяє проводити моделювання роботи автомату у різних сценаріях, що допомагає передбачити доходи при зміні попиту, збільшенні кількості автоматів чи підвищенні вартості продукту.

Аналіз та звітність. Система автомату може надавати регулярні звіти про кількість проданих рідин, енерговитрати та необхідність обслуговування. Ці дані використовуються для точного аналізу прибутковості та оптимізації роботи.

Оцінка економічної ефективності системи автомату з продажу рідин базується на аналізі початкових інвестицій, експлуатаційних витрат та прогнозованої прибутковості. Застосування сучасних технологій для оптимізації витрат, таких як енергоефективні рішення та автоматизація процесів обслуговування, дозволяє значно підвищити економічну ефективність системи та скоротити терміни її окупності.

## 5.2 Порівняльний аналіз з аналогічними рішеннями на ринку

Процес розробки та впровадження автомату з продажу рідин вимагає врахування ряду ключових витрат на кожному етапі життєвого циклу системи. Ця оцінка допоможе визначити фінансові ресурси, необхідні для реалізації проекту, а також оптимізувати бюджет для досягнення максимальної рентабельності.

### 1. Розробка апаратного забезпечення

Першим етапом є розробка та закупівля апаратного забезпечення, що є основою для функціонування автомату. Витрати включають:

Контролер і процесор: основний елемент для управління роботою автомату. Залежно від обраної моделі.

Сенсори та актори: компоненти для вимірювання рівня рідини, контролю температури, відкриття/закриття клапанів тощо. Сюди входять ультразвукові датчики, інфрачервоні сенсори, клапани та насоси.

Дисплей та інтерфейс користувача: для забезпечення зручного інтерфейсу користувача, необхідно придбати дисплей, клавіатуру або

сенсорний екран.

Корпус та механічні елементи: спеціальний корпус для захисту внутрішніх компонентів автомату, а також механічні частини для системи дозування.

## 2. Розробка програмного забезпечення

Програмне забезпечення є ключовою складовою роботи автомату, що забезпечує управління апаратною частиною, обробку платежів та інтеграцію з іншими системами. Основні витрати на розробку ПЗ включають:

Інтеграція з датчиками та акторами: написання коду для управління всіма сенсорами та виконавчими механізмами. Витрати на розробку можуть включати зарплату програміста або вартість послуг зовнішніх розробників.

Обробка платежів: розробка програмного модуля для прийому платежів через банківські картки або мобільні платіжні системи.

Система управління та моніторингу: розробка інтерфейсу для адміністратора автомату, що дозволить контролювати його роботу, переглядати статистику та проводити технічне обслуговування

## 3. Інсталяція та налаштування

Наступний етап включає інсталяцію автомату на місці експлуатації, підключення до мережі живлення, водопостачання та комунікаційних систем. Витрати включають:

Вартість монтажу: встановлення автомату в приміщенні або на відкритому просторі. Ці витрати включають оплату праці монтажників і техніків.

Калібрування та налаштування: необхідно провести початкове калібрування датчиків та налаштування дозувальних систем для забезпечення точного вимірювання рідини.

#### 4. Технічне обслуговування та підтримка

Для забезпечення безперебійної роботи автомату необхідно передбачити витрати на регулярне обслуговування та технічну підтримку, що включають:

Періодичне технічне обслуговування: регулярне очищення компонентів автомату, заміна зношених деталей та перевірка функціональності системи.

Оновлення програмного забезпечення: періодичне оновлення ПЗ для виправлення помилок, покращення функціональності та інтеграції з новими платіжними системами.

#### 5. Витрати на сертифікацію та ліцензування

Перед впровадженням автомату з продажу рідин необхідно отримати відповідні сертифікати якості та ліцензії, що дозволяють використовувати систему на ринку. Ці витрати можуть включати:

Сертифікація безпеки обладнання: підтвердження відповідності автомату вимогам безпеки для користувачів.

Ліцензування платіжних систем: для інтеграції з платіжними системами необхідно отримати відповідні ліцензії на обробку транзакцій.

#### 6. Маркетинг та просування

Для успішного впровадження автомату на ринок необхідно провести рекламні кампанії та популяризувати продукт серед потенційних користувачів:

Рекламні витрати: проведення маркетингових кампаній, створення рекламних матеріалів та презентацій автомату на різних виставках і конференціях.

Оцінка вартості розробки та впровадження автомату з продажу рідин є важливим етапом планування проекту. Ретельний аналіз всіх етапів, від апаратної та програмної розробки до інсталяції та технічного обслуговування, дозволяє забезпечити економічну ефективність та мінімізувати ризики при впровадженні системи.

### 5.3 Перспективи розвитку та вдосконалення

Автомати для продажу рідин є технологічно перспективним напрямком, який має значний потенціал для подальшого розвитку. Завдяки впровадженню новітніх технологій у програмування, апаратне забезпечення та систему взаємодії з користувачем, автоматизовані системи можуть бути значно вдосконалені, що підвищить їх продуктивність, зручність та функціональність.

#### 1. Інтеграція з IoT

Однією з найбільш перспективних технологій для подальшого розвитку автоматів є Інтернет речей (IoT). Впровадження IoT дозволить автоматам передавати дані про стан обладнання, обсяг продажу, необхідність технічного обслуговування та запаси рідин у реальному часі до хмарних систем управління. Це забезпечить можливість віддаленого моніторингу та контролю, зменшить час на обслуговування та знизить операційні витрати.

#### 2. Машинне навчання та прогнозування попиту

Застосування алгоритмів машинного навчання для автоматів з продажу рідин дозволить аналізувати поведінку користувачів та



прогнозувати попит на рідини у різних місцях. Це дасть можливість оптимізувати процес поповнення запасів та більш ефективно планувати роботу автоматів. Наприклад, алгоритми можуть визначати періоди пікового попиту, що дозволить уникнути нестачі продукту або перевантаження системи.

### 3. Розширення можливостей платіжних систем

На сьогоднішній день більшість автоматів використовують стандартні банківські карти для оплати. Проте розвиток технологій оплати дозволяє впроваджувати нові способи, такі як мобільні платіжні системи (Apple Pay, Google Pay), криптовалютні платежі або навіть біометричну аутентифікацію. Розширення платіжних можливостей підвищить зручність для користувачів та знизить залежність від одного способу оплати.

### 4. Підвищення енергоефективності

Одним із напрямків вдосконалення є підвищення енергоефективності автоматів. Впровадження енергоощадних компонентів, таких як енергоефективні процесори, датчики та актори, дозволить знизити споживання електроенергії. Крім того, можна впроваджувати системи автоматичного відключення невикористовуваних частин обладнання, коли автомат знаходиться в режимі очікування.

### 5. Підвищення безпеки та захисту даних

Оскільки автомати працюють з особистими даними користувачів та обробляють фінансові транзакції, важливим напрямком є вдосконалення систем безпеки. Використання сучасних методів шифрування, багатофакторної аутентифікації та захисту від кібератак забезпечить

більшу безпеку даних. Це особливо актуально у випадках, коли автомат підключений до Інтернету для роботи з платіжними системами чи віддаленого моніторингу.

#### 6. Персоналізація взаємодії з користувачем

Інтерфейс автоматів може бути значно покращений за допомогою персоналізації послуг. Застосування штучного інтелекту дозволить автоматом розпізнавати користувачів, надавати індивідуальні пропозиції та враховувати попередні покупки для рекомендації нових товарів. Це може підвищити рівень задоволеності клієнтів і стимулювати повторні покупки.

#### 7. Розширення асортименту продуктів

Існуючі автомати часто обмежені у виборі рідин, які вони можуть продавати. Розширення можливостей автоматів, зокрема через вдосконалення систем дозування та розливу різних типів рідин, дозволить пропонувати більший асортимент продуктів. Автомати можуть адаптуватися до продажу різних напоїв, від газованих вод до кави та інших видів рідин.

#### 8. Автономність та робота на сонячних батареях

А Автомати, що працюють у віддалених або важкодоступних місцях, можуть бути оснащені сонячними батареями для забезпечення енергії. Це зробить їх повністю автономними та зменшить витрати на електроживлення. Поєднання сонячної енергії з енергоощадними технологіями дозволить підвищити ефективність таких систем.

#### 9. Використання екологічно чистих матеріалів

З огляду на підвищену увагу до екології, автомати можуть бути

розроблені з використанням матеріалів, що піддаються переробці, або біорозкладних елементів. Це сприятиме зменшенню впливу на навколишнє середовище та відповідатиме тенденціям сталого розвитку.

Перспективи розвитку автоматів для продажу рідин включають впровадження новітніх технологій, таких як IoT, машинне навчання, вдосконалені платіжні системи та персоналізація інтерфейсу. Крім того, підвищення енергоефективності, безпеки та екологічності дозволить зробити ці системи більш конкурентоспроможними та привабливими для користувачів.

## ВИСНОВКИ

Під час дослідження, яке було проведене в рамках кваліфікаційної роботи на тему "Методи і технології програмування автоматів з продажу рідин", було розглянуто основні аспекти проектування, розробки та впровадження програмних систем для автоматизованих рішень продажу рідин. В результаті виконаної роботи можна зробити наступні ключові висновки:

Аналіз методів програмування показав, що для успішного функціонування автоматів з продажу рідин важливо забезпечити надійну архітектуру системи, яка включає функціональні модулі для керування сенсорами, акторами, обробки платежів та дозування рідин. Використання модульного підходу дозволяє легко масштабувати систему та інтегрувати нові функції.

Обґрунтовано вибір апаратного забезпечення, яке відповідає вимогам до роботи автоматів. Програмні рішення, описані в роботі, інтегровані з сучасними мікроконтролерами, датчиками рівня рідини, платіжними системами та іншими компонентами, що забезпечують надійну і стабільну роботу автомату.

Алгоритми роботи автомату було розроблено з урахуванням різних сценаріїв взаємодії користувача з системою. Ключовими процесами стали дозування рідини, обробка платежів, контроль запасів та індикація помилок. Усі ці процеси побудовані на основі асинхронного оброблення подій, що забезпечує стабільність системи навіть при високих навантаженнях.

Система контролю оплати та розрахунку вартості побудована з урахуванням сучасних вимог до безпеки транзакцій. Інтеграція з

платіжними терміналами забезпечує можливість приймання різних типів оплат, включаючи безконтактні платежі. Також розроблена система захисту від шахрайства та помилок у розрахунках.

Точність дозування та видачі рідин було забезпечено завдяки використанню сенсорів рівня рідини та високоточних дозаторів. Це дозволило мінімізувати витрати рідин і підвищити точність кожної операції, що є важливим критерієм для ефективної роботи автомату.

Тестування програмного забезпечення показало стабільність і ефективність роботи системи у різних режимах. Для перевірки працездатності були змодельовані різні сценарії роботи автомату, включаючи ситуації з нестачею рідини, збої у платіжних системах та інші критичні ситуації. Система успішно впоралася з усіма завданнями.

Забезпечення безпеки та захист даних було реалізовано за допомогою впровадження шифрування даних, автентифікації користувачів та захисту від несанкціонованого доступу. Це дозволило мінімізувати ризики втручання у роботу автомату та захистити фінансові дані користувачів.

Економічна ефективність системи була оцінена з урахуванням вартості розробки та впровадження, витрат на апаратне забезпечення та потенційних доходів від роботи автоматів. Показано, що автоматизація процесів продажу рідин дозволяє значно скоротити операційні витрати та підвищити рентабельність бізнесу.

Перспективи вдосконалення автоматів з продажу рідин включають впровадження новітніх технологій, таких як Інтернет речей (IoT), штучний інтелект для прогнозування попиту, інтеграція альтернативних платіжних систем, підвищення енергоефективності та використання екологічних матеріалів.

Таким чином, виконане дослідження показало, що розроблені методи та технології програмування автоматів з продажу рідин є ефективними і можуть бути успішно застосовані в сучасних автоматизованих системах. Впровадження таких автоматів сприятиме підвищенню зручності для споживачів, а також підвищить ефективність бізнес-процесів для підприємств.

Також у результаті проведеного дослідження на тему "Методи і технології програмування автоматів з продажу рідин" було виявлено, що розроблені системи та алгоритми функціонують ефективно, проте існують можливості для подальшого вдосконалення. Нижче наведено ряд рекомендацій, які можуть покращити роботу автоматів та їхню інтеграцію у ринок:

Покращення інтерфейсу користувача:

Розробка більш інтуїтивно зрозумілого інтерфейсу, що включає в себе можливість персоналізації для різних типів користувачів.

Впровадження мультимедійних елементів, таких як відеоінструкції або анімації, може допомогти в орієнтації користувачів.

Розширення платіжних можливостей:

Інтегрувати нові платіжні методи, такі як криптовалюти, системи лояльності та мобільні гаманці, що підвищить доступність автоматів для більшої кількості споживачів.

Впровадити функцію розрахунку під час підходу до автомату (наприклад, за допомогою QR-кодів).

Системи моніторингу і прогнозування:

Реалізувати механізми збору даних про споживчі звички, які можуть допомогти у прогнозуванні попиту на певні рідини.

Впровадження аналітичних інструментів для моніторингу продажів і управління запасами в режимі реального часу.

Покращення точності дозування:

Впровадження вдосконалених сенсорів для забезпечення точності дозування рідин, що зменшить відходи та підвищить довіру споживачів до якості продукту.

Регулярне калібрування дозувальних механізмів для підтримки високої точності видачі.

Екологічні ініціативи:

Розглянути можливість використання екологічно чистих матеріалів у конструкції автоматів та упаковці рідин.

Запровадження системи збору відходів або вторинної переробки упаковки рідин, що сприятиме покращенню іміджу компанії та її соціальної відповідальності.

Покращення енергетичної ефективності:

Використання енергозберігаючих технологій, які зменшують витрати електроенергії під час роботи автомату, що може суттєво знизити експлуатаційні витрати.

Інтеграція сонячних панелей для живлення частини елементів автомату.

Дослідження нових технологій:

Регулярно слідкувати за новими технологічними тенденціями в галузі автоматизації та програмування, щоб своєчасно впроваджувати інновації.

Партнерство з технологічними компаніями може прискорити

процес впровадження нових рішень.

Ці рекомендації не тільки допоможуть підвищити ефективність та зручність роботи автоматів з продажу рідин, але й забезпечать стійкий розвиток бізнесу в умовах конкурентного ринку. Реалізація запропонованих заходів дозволить створити більш привабливий продукт для споживачів і сприятиме розвитку нових ринкових ніш.



## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

Боярчук А. І., Ковальчук М. В., Якимович О. О. Автоматизовані системи керування процесами: технології та застосування. — Київ: Видавничий центр "Освіта", 2018.

Іванченко О. М., Костюк С. М. Інформаційні технології в автоматизації виробничих процесів. — Харків: ХНУРЕ, 2020.

Попов В. П., Дрогомирецький С. М. Автомати з продажу товарів та послуг: технічні рішення і перспективи розвитку. — Львів: Видавництво "Техніка", 2019.

Захаров О. В. Програмне забезпечення вбудованих систем: принципи та підходи. — Харків: ХАІ, 2017.

Сучасні методи ідентифікації користувачів в автоматизованих системах / Під ред. О. В. Сергієнка. — Київ: Академія, 2021.

Науменко В. М. Інноваційні рішення в галузі автоматизованих торгових систем. // Журнал "Інформаційні технології та автоматизація", № 3, 2021.

Smith J., Brown L. Embedded Systems for Vending Machines: Design and Programming Approaches. — New York: McGraw-Hill, 2019.

Li Y., Chang P. Smart Vending Machines: IoT and Data-Driven Solutions. // International Journal of Industrial Electronics and Automation, Vol. 12, Issue 2, 2020.

Коваленко І. В., Гончарук С. А. Інтегровані платіжні системи для автоматів з продажу рідин. — Київ: НТУУ "КПІ", 2020.

Збірник наукових праць "Автоматизація процесів продажу рідин: теорія і практика". — Київ: Інститут автоматички, 2022.

ISO/IEC 27001:2013. Information Security Management Systems — Requirements. International Organization for Standardization, 2013.

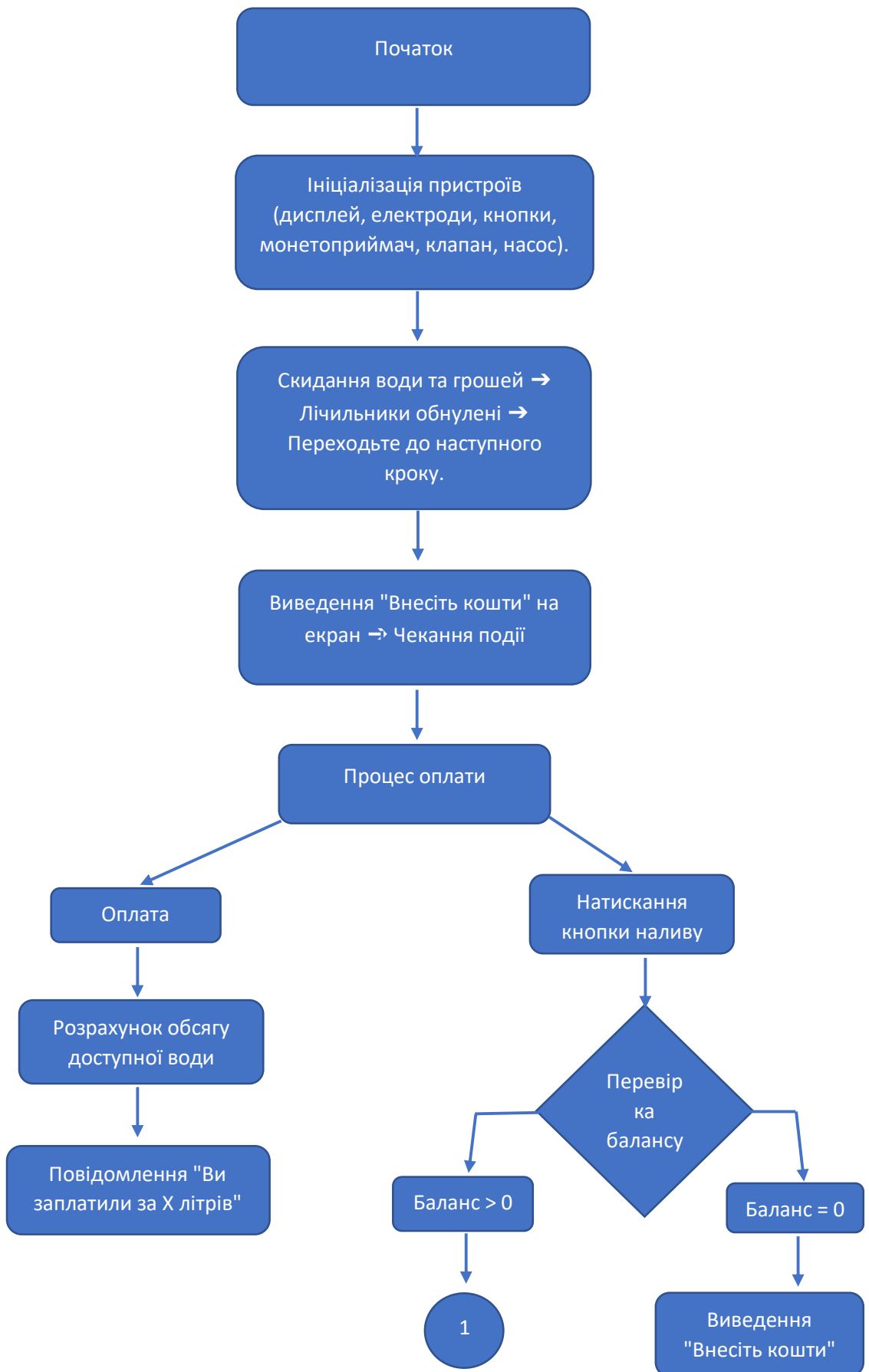
Кравченко П. І. Енергозберігаючі технології в автоматизованих системах. — Вінниця: ВНТУ, 2021.

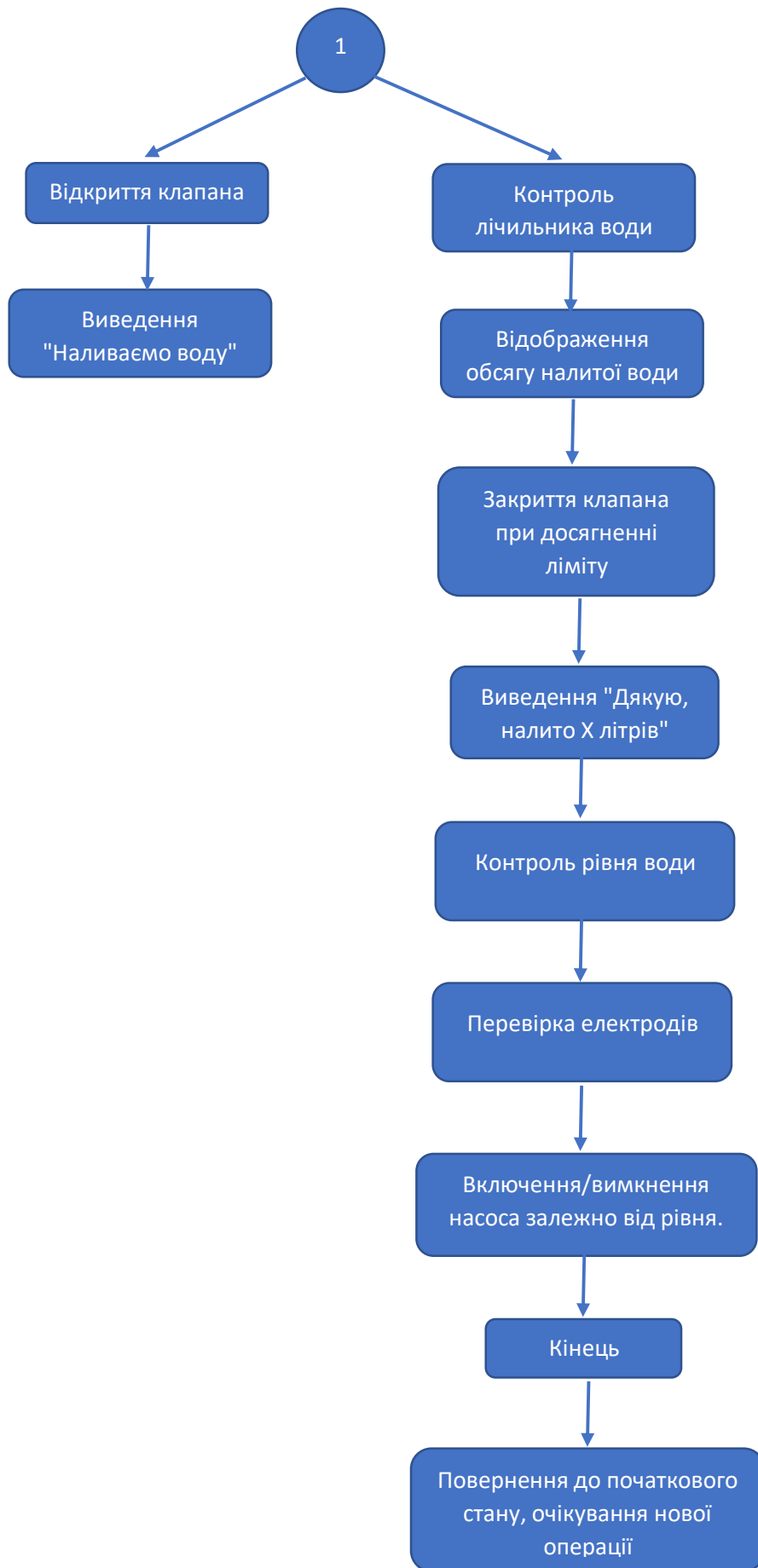
Системи моніторингу стану обладнання автоматів з продажу товарів / Укл. В. В. Михайлов, П. О. Степаненко. — Одеса: ОНАХТ, 2020.

Robles G., Hernandez L. Vending Machine Technologies: Security and Data Protection. — London: Springer, 2018.

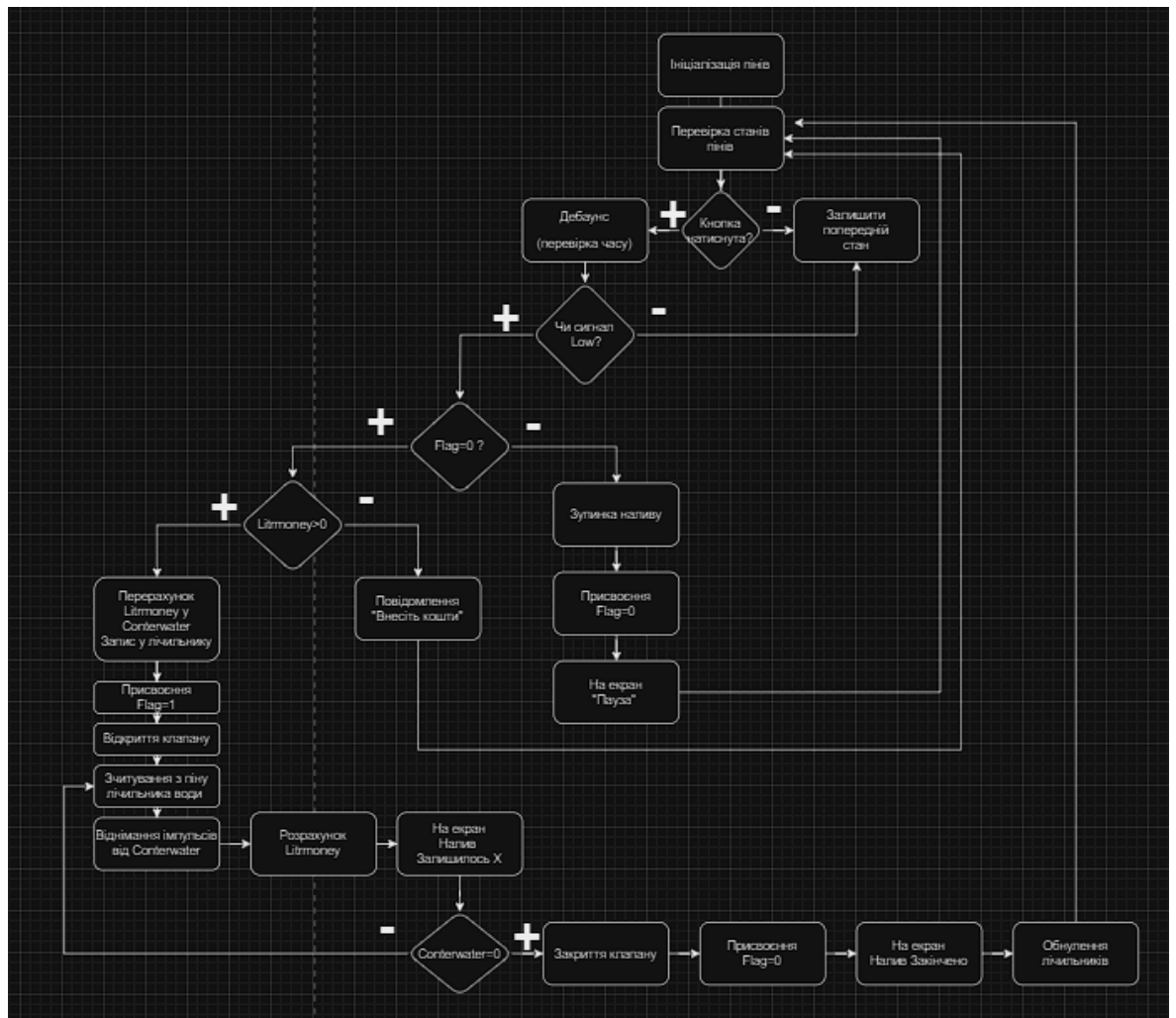
Олексійчук М. В. Системи управління сенсорами та акторами для торгових автоматів. // Журнал "Сучасні проблеми автоматизації", № 4, 2020.

## Додаток А. Алгоритм опрацювання системи автомату

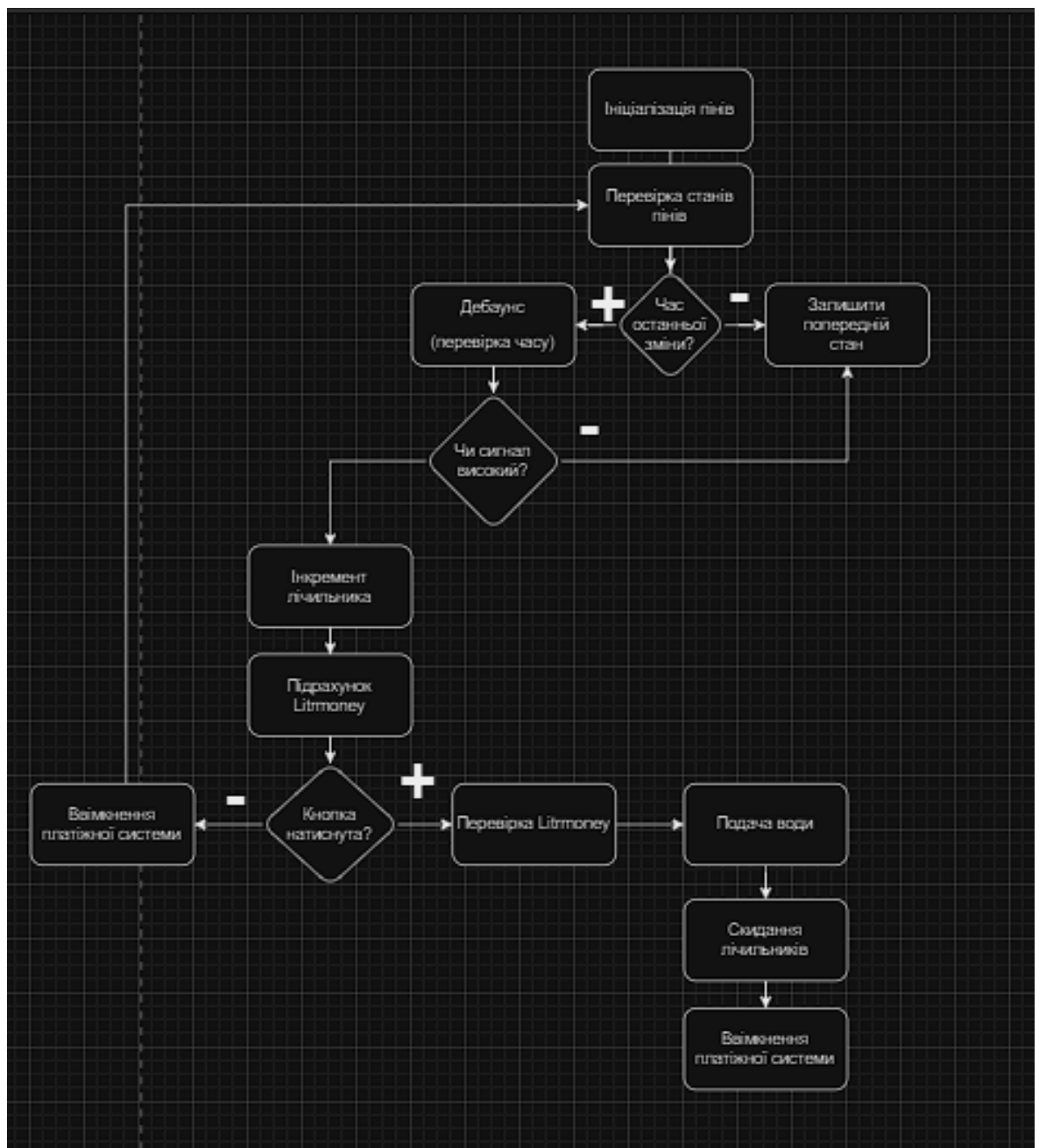




## Додаток Б. Алгоритм опрацювання кнопки користувача



## Додаток В. Алгоритм опрацювання платіжної системи



## Додаток Г. Програмний код

```
#include <LiquidCrystal.h> // Налив перевірити
LiquidCrystal lcd(14, 15, 16, 17, 18, 19); // мій LCD ініціалізується так
byte f0[8] = {                                     // бітова маска виводу Z

    B01110,
    B10001,
    B00001,
    B00110,
    B00001,
    B00001,
    B10001,
    B01110,

};

byte f1[8] = {                                     // бітова маска виводу Л

    0b00111,
    0b01001,
    0b10001,
    0b10001,
    0b10001,
    0b10001,
    0b10001,
    0b00000,

};

byte f2[8] = {                                     // бітова маска виводу У

    0b10001,
    0b10001,
```

```

0b01001,
0b00111,
0b00001,
0b00001,
0b00010,
0b11100, } ;
byte f3[8] = { //бітова маска виводу В
                В11110,
                В10001,
                В10001,
                В11110,
                В10001,
                В10001,
                В10001,
                В11110,

};
byte f5[8] = { //бітова маска виводу В
                В10001,
                В10001,
                В10001,
                В11001,
                В10101,
                В10101,
                В10101,
                В11001,

};

```



```
byte f6[8] = { //бітова маска виводу Д
               B01110,
               B01010,
               B01010,
               B01010,
               B01010,
               B01010,
               B01010,
               B11111,
               B10001,
```

```
};
```

```
byte f7[8] = { //бітова маска виводу П
               Ob11111,
               Ob10001,
               Ob10001,
               Ob10001,
               Ob10001,
               Ob10001,
               Ob10001,
               Ob10001,
               Ob00000,
```

```
};
```

```
byte f12[8] = { //бітова маска виводу И
               Ob10001,
               Ob10001,
               Ob10011,
               Ob10101,
               Ob11001,
               Ob10001,
               Ob10001,
```

```

0b000000,
};
//.....Запуск раз в 600 секунд .....
int timer = LOW; //цією перемінною виставляємо стан таймера
long timerMillis = 0; // зберігаємо час останнього таймера
long timerinterval = 120000; //інтервал між включенням-виключенням таймеру
unsigned long timercurrentMillis = 0;

int temperat;

const int ellow= 0;    // електрод нижній чистої води
const int elhigh = 1; // електрод верхній чистої води
const int buttonwater = 3;    // Кнопка наливу води
const int coinbanknote = 4;    // Монетоприймач и Купюроприймач
const int Pinwater = 5;    // Номер піна до якого підключений лічильник води
const int valveissue= 6;    // Клапан видачі води
const int pump = 13;    // насос вхідної води

float counterwater = 0;    // лічильник імпульсів води
int dialHasFinishedRotatingAfterMsWater = 10;    // лічильник імпульсів води по закінченню 10 Ms припиняє прийом імпульсів
int debounceDelayWater = 1;    // лічильник імпульсів води імпульси менше 15 Ms не рахуються
long lastStateChangeTimeWater = 0;    // вимірюємо час
int needToPrintWater = 0;
int lastStateWater = LOW;    //лічильник імпульсів води стан в минулому
int trueStateWater = LOW;    //лічильник імпульсів води стан в теперішньому

```

```

    int buttonwaterState= LOW; // стан кнопки
наливу води якщо HIGH вода наливається якщо LOW слічильник
імпульсів води менше або дорівнює нулю вода зупинена.

    float countmoney=0.000; // лічильник грошей
    float Litrmoney=0.000; //гроші помножені на 355
    long lastStateChangeTime = 0; // вимірюємо час
    int needToPrint = 0;

    int buttonClearD;

int dialHasFinishedRotatingAfterMs = 100; // по закінченню 100 Ms
припиняємо прийом імпульсів
int debounceDelay = 10; // імпульси менше 10 Ms не рахуються
int lastState = LOW; //стан в минулому
int trueState = HIGH; //стан в теперішньому
    int buttonwaterD;

//додаємо змінні для нової кнопки
int flag=0;
int flag9=0;
//10 коп
int needToPrint1 = 0;
int count1;
int in1 = 8;
int reading1;
int lastState1 = LOW;
int trueState1 = LOW;
long lastStateChangeTime1 = 0;
int cleared1 = 0;
int dialHasFinishedRotatingAfterMs1 = 200;
int debounceDelay1 = 10;
//50 коп
int needToPrint10 = 0;

```

```
int count10;
int in10 = 10;
int reading10;
int lastState10 = LOW;
int trueState10 = LOW;
long lastStateChangeTime10 = 0;
int cleared10 = 0;
int dialHasFinishedRotatingAfterMs10 = 200;
int debounceDelay10 = 10;
// 1 грн
int needToPrint11 = 0;
int count11;
int in11 = 11;
int reading11;
int lastState11 = LOW;
int trueState11 = LOW;
long lastStateChangeTime11 = 0;
int cleared11 = 0;
int dialHasFinishedRotatingAfterMs11 = 200;
int debounceDelay11 = 10;
void setup()
{
    // Serial.begin(9600);
    pinMode(buttonwater, INPUT_PULLUP); // кнопка наливу води
    pinMode(4, INPUT_PULLUP); // купюроприймач
    pinMode(5, INPUT_PULLUP); // Номер піна до якого підключений
лічильник води
    pinMode(6, OUTPUT); //Клапан видачі води
    pinMode(8, INPUT_PULLUP); // 10 коп
```

```

pinMode(10, INPUT_PULLUP); // 50 коп
pinMode(11, INPUT_PULLUP ); //1 грн
digitalWrite(valveissue, LOW); //pin 6
pinMode(0, INPUT); //електрод нижній чистої води
pinMode(1, INPUT); //електрод верхній чистої води
pinMode(13, OUTPUT);//запрограмована на введення
lcd.createChar(0, f0); //створюємо літеру З
lcd.createChar(1, f1); //створюємо літеру Л
lcd.createChar(2, f2); //створюємо літеру У
lcd.createChar(3, f3); //створюємо літеру В
lcd.createChar(5, f5); //створюємо літеру Ы
lcd.createChar(6, f6); //створюємо літеру Д
lcd.createChar(7, f7); //створюємо літеру П
lcd.createChar(12, f12); //створюємо літеру И
lcd.begin(16, 2);
lcd.clear();
}
void loop()
{
// обнуляємо автоматично на протязі 10 секунд після включення гроші і
лічильник води
if(millis())<10000)
{
counterwater = 0;
countmoney=0;
Litrmoney=1;
//Заплатите
lcd.setCursor(0, 0);
lcd.clear();

```

```

    lcd.write(byte(0)); lcd.print("A"); lcd.write(byte(7));
lcd.write(byte(1));lcd.print("A");lcd.print("T");lcd.write(byte(12));lcd.print("
T");lcd.print("E");

    //За воду

    lcd.setCursor(9, 1);

    lcd.write(byte(0)); lcd.print("A");

    lcd.setCursor(12, 1);

    lcd.write(byte(3)); lcd.print("O"); lcd.write(byte(6)); lcd.write(byte(2));
}

//.....кнопка нова

if(digitalRead(3)== LOW && flag==0 && countmoney != 0 &&
Litrmoney-counterwater >= 0)//если кнопка нажата
{
    if (digitalRead(6)==1)
    {
        // Налито літрів

        lcd.clear();

        lcd.setCursor(0, 0);

        lcd.write(byte(3));

        lcd.write(byte(5));

        lcd.setCursor(5, 0);

        lcd.print("H"); lcd.print("A"); lcd.write(byte(1));lcd.write(byte(12));
lcd.print("T");lcd.print("O");

        lcd.setCursor(3, 1);

        lcd.write(byte(0));

        lcd.print("A ");

        lcd.print(countmoney);

        lcd.print(counterwater/355);

        lcd.setCursor(9, 1);

        lcd.write(byte(1)); lcd.write(byte(12));lcd.print("T"); lcd.print("P");

```

```

    lcd.print("O");
    lcd.write(byte(3));

}
if (digitalRead(6)==0)
{
    // Наливаємо Воду
    lcd.clear();
    lcd.setCursor(3, 0);
    lcd.print("H"); lcd.print("A"); lcd.write(byte(1)); lcd.write(byte(12));
    lcd.write(byte(3)); lcd.print("A"); lcd.print("E"); lcd.print("M");
    lcd.setCursor(5, 1);
    lcd.write(byte(3)); lcd.print("O"); lcd.write(byte(6)); lcd.write(byte(2));
}
flag=1;
}
if(digitalRead(3)== HIGH )//якщо кнопка НЕ натиснута
{
    flag=0;//обнуляємо змінну flag
}
// працюємо з лічильником води
int readingwater = digitalRead(Pinwater);
if ((millis() - lastStateChangeTimeWater) >
dialHasFinishedRotatingAfterMsWater) {
    if (needToPrintWater) {
        needToPrintWater = 0;

    }
}
}

```

```
if (readingwater != lastStateWater) { //логічне заперечення НІ минулий стан не відповідає теперішньому
```

```
    lastStateChangeTimeWater = millis(); //привласнення змінної lastStateChangeTime теперішнє значення millis
```

```
}
```

```
if ((millis() - lastStateChangeTimeWater) > debounceDelayWater)
```

```
{
```

```
    if (readingwater != trueStateWater) {
```

```
        trueStateWater = readingwater;
```

```
        if (trueStateWater == HIGH) {
```

```
            // збільшуємо лічильник на 1, якщо стан піна HIGH
```

```
            counterwater++;
```

```
            needToPrintWater = 1; // необхідно вивести підсумкову цифру по закінченню обертання диску лічильника води
```

```
        }
```

```
    }
```

```
}
```

```
lastStateWater = readingwater;
```

```
if ( (Litrmoney-counterwater) <= 0 )
```

```
{
```

```
    digitalWrite(valveissue, LOW); // відключає клапан подачі води
```

```
        digitalWrite(foodbillacceptor, HIGH); // включає купюроприймач
```

```
        Litrmoney=1;
```

```
        counterwater = 0;
```

```
        countmoney=0;
```

```
    //Заплатите
```

```
    lcd.clear();
```

```
    lcd.setCursor(0, 0);
```



```

    lcd.write(byte(0)); lcd.print("A"); lcd.write(byte(7));
lcd.write(byte(1));lcd.print("A");lcd.print("T");lcd.write(byte(12));lcd.print("
T");lcd.print("E");

    //За воду

    lcd.setCursor(9, 1);

    lcd.write(byte(0)); lcd.print("A");

    lcd.setCursor(12, 1);

    lcd.write(byte(3)); lcd.print("O"); lcd.write(byte(6)); lcd.write(byte(2));
}

int reading = digitalRead(coinbanknote);

if ((millis() - lastStateChangeTime) > dialHasFinishedRotatingAfterMs) {
    // Цифра не набирається, або не зчитується останній імпульс
    if (needToPrint) {
        // ви заплатили за літрів

        lcd.clear();

        lcd.setCursor(0, 0);

        lcd.write(byte(3)); lcd.write(byte(5));

        lcd.setCursor(3, 0);

        lcd.write(byte(0)); lcd.print("A"); lcd.write(byte(7)); lcd.write(byte(1));
lcd.print("A"); lcd.print("T");lcd.write(byte(12));
lcd.write(byte(1));lcd.write(byte(12));

        lcd.setCursor(1, 1);

        lcd.write(byte(0)); lcd.print("A "); lcd.print(countmoney/1.4);

        lcd.setCursor(10, 1);

        lcd.write(byte(1)); lcd.write(byte(12));lcd.print("T"); lcd.print("P");
lcd.print(".");

        needToPrint = 0;
    }
}
}

```

```

    if (reading != lastState) { //логічне заперечення НІ минулий стан не
відповідає теперішньому

        lastStateChangeTime = millis(); //привласнення змінної
lastStateChangeTime теперішнє значення millis

    }

    if ((millis() - lastStateChangeTime) > debounceDelay) {

        if (reading != trueState) {

            trueState = reading;

            if (trueState == HIGH) { //збільшуємо лічильник на 1, якщо стан піна
HIGH

                countmoney++;

                countmoney++;

                Litrmoney=(countmoney/1.4*355);

                needToPrint = 1; // необхідно вивести підсумкову цифру по
закінченню обертання диску лічильника води

            }

        }

    }

    lastState = reading;

    // 10 коп

    reading1 = digitalRead(8);

    if ((millis() - lastStateChangeTime1) > dialHasFinishedRotatingAfterMs1) {

        if (needToPrint1) {

            needToPrint1 = 0;

            count1 = 0;

            cleared1 = 0;

        }

    }

    if (reading1 != lastState1) {

        lastStateChangeTime1 = millis();

```

```

}
if ((millis() - lastStateChangeTime1) > debounceDelay1) {
if (reading1 != trueState1) {
trueState1 = reading1;
if (trueState1 == HIGH) {
count1++;
countmoney=countmoney+0.2;
Litrmoney=(countmoney/1.4*355);
    // Вы Заплатили за Литров
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.write(byte(3)); lcd.write(byte(5));
    lcd.setCursor(3, 0);
    lcd.write(byte(0)); lcd.print("A"); lcd.write(byte(7)); lcd.write(byte(1));
    lcd.print("A"); lcd.print("T");lcd.write(byte(12));
    lcd.write(byte(1));lcd.write(byte(12));
    lcd.setCursor(1, 1);
    lcd.write(byte(0)); lcd.print("A "); lcd.print(countmoney/1.4);
    lcd.setCursor(10, 1);
    lcd.write(byte(1)); lcd.write(byte(12));lcd.print("T");
    lcd.print("P");lcd.print(".");
    needToPrint1 = 1;
}
}
}
lastState1 = reading1;
//50 коп
reading10 = digitalRead(10);
if ((millis() - lastStateChangeTime10) > dialHasFinishedRotatingAfterMs10)
{

```

```

if (needToPrint10)
{
needToPrint10 = 0;
count10 = 0;
cleared10 = 0;
}
}

if (reading10 != lastState10) {
lastStateChangeTime10 = millis();
}

if ((millis() - lastStateChangeTime10) > debounceDelay10) {
if (reading10 != trueState10) {
trueState10 = reading10;
if (trueState10 == HIGH) {
count10++;
countmoney=countmoney+1;
Litrmoney=(countmoney/1.4*355);
// Вы Заплатили за Литров
lcd.clear();
lcd.setCursor(0, 0);
lcd.write(byte(3)); lcd.write(byte(5));
lcd.setCursor(3, 0);
lcd.write(byte(0)); lcd.print("A"); lcd.write(byte(7)); lcd.write(byte(1));
lcd.print("A"); lcd.print("T");lcd.write(byte(12));
lcd.write(byte(1));lcd.write(byte(12));
lcd.setCursor(1, 1);
lcd.write(byte(0)); lcd.print("A "); lcd.print(countmoney/1.4);
lcd.setCursor(10, 1);
lcd.write(byte(1)); lcd.write(byte(12));lcd.print("T"); lcd.print("P");
lcd.print(".");

```

```

needToPrint10 = 1;
}
}
}
lastState10 = reading10;
// 1 грн.
reading11 = digitalRead(11);
if ((millis() - lastStateChangeTime11) > dialHasFinishedRotatingAfterMs11)
{
if (needToPrint11)
{
needToPrint11 = 0;
count11 = 0;
cleared11 = 0;
}
}
if (reading11 != lastState11) {
lastStateChangeTime11 = millis();
}
if ((millis() - lastStateChangeTime11) > debounceDelay11) {
if (reading11 != trueState11) {
trueState11 = reading11;
if (trueState11 == HIGH) {
count11++;
countmoney=countmoney+2;
Litrmoney=(countmoney/1.4*355);
// Вы Заплатили за Литров
lcd.clear();
lcd.setCursor(0, 0);

```

```

    lcd.write(byte(3)); lcd.write(byte(5));

    lcd.setCursor(3, 0);

    lcd.write(byte(0)); lcd.print("A"); lcd.write(byte(7)); lcd.write(byte(1));
    lcd.print("A"); lcd.print("T");lcd.write(byte(12));
    lcd.write(byte(1));lcd.write(byte(12));

    lcd.setCursor(1, 1);

    lcd.write(byte(0)); lcd.print("A "); lcd.print(countmoney/1.4);

    lcd.setCursor(10, 1);

    lcd.write(byte(1)); lcd.write(byte(12));lcd.print("T"); lcd.print("P");
    lcd.print(".");
    needToPrint11 = 1;
}
}
}
lastState11 = reading11;
// працюємо з чистим баком
if (digitalRead(ellow) == LOW && digitalRead(elhigh) == LOW)
{
    digitalWrite(pump, LOW);// відключає насос
}
if (digitalRead(ellow) == HIGH && digitalRead(elhigh) == HIGH)
{
    digitalWrite(pump, HIGH);// Включає насос
}
}

```