АНОТАЦІЯ

наукової роботи під шифром “Контроль”

Дана наукова робота містить огляд сучасних систем і методів біометричної ідентифікації особи. В роботі розглянуті різноманітні принципи роботи і функціонування систем біометричної ідентифікації. Розглянуто методи порівняння відбитків пальців, принцип роботи сканерів відбитків пальців і їх види. Розглянуті різноманітні атаки і загрози на біометричні системи.

У роботі розроблено прототип системи біометричної ідентифікації на базі мікроконтролера Arduino Uno. Наведено детальний опис функціонування даної системи, розроблено програмне забезпечення. Проведені дослідження якості і надійності розробленої системи. Наведено результати даних досліджень.

Загальна характеристика наукової роботи. Робота містить: вступ, три розділи, висновки, список використаної літератури. Кількість сторінок – 30, рисунків – 18, таблиць – 3, використаних наукових джерел – 11, додатків – 5.

Ключові слова: відбиток пальця, біометрія, ідентифікація.

Шифр “Контроль”

Система контролю доступу на основі біометричних ідентифікаторів

ЗМІСТ

ВСТУП

В наш час для ідентифікації особи використовують не тільки відбитки пальців, а й райдужну оболонку ока, обличчя, голос, форму вуха, відбиток долоні, почерк, підпис і інші біометричні дані людини, які є унікальними. Системи контролю доступу на основі біометричних ідентифікаторів забезпечують досить якісний і надійний захист інформації. Вони вже мають досить широке розповсюдження в сучасному світі, але не перестають розвиватися і удосконалюватися. Як і будь-які системи, вони мають низку переваг і недоліків, слабкі місця і можливі недопрацювання. Але в той же час, дана тема є досить перспективною, цікавою для вивчення і вдосконалення.

Дана робота є актуальною з точки зору розробки недорогої але надійної біометричної системи контролю доступу, яка має безліч варіацій для практичного використання.

Метою наукової роботи є дослідження технологій біометричної ідентифікації, розробка і створення прототипу системи біометричної ідентифікації.

Об’єкт дослідження: методи ідентифікації людини.

Предмет дослідження: застосування відбитків пальців людини як біометричного ідентифікатора.

Для досягнення поставленої мети в роботі розв’язуються наступні задачі.

1. Здійснити пошук інформації про біометричні ідентифікатори людини, системи біометричної ідентифікації, технічні засоби реалізації даних систем, провести їх оцінку і порівняння.
2. Розробити працюючий прототип системи контролю доступу на основі біометричного ідентифікатора. Розробити програмне забезпечення, представити практичну реалізацію розробленої системи.
3. Провести дослідження розробленої системи, дати оцінку її ефективності і надійності.

РОЗДІЛ 1. ПРИНЦИПИ ФУНКЦІОНУВАННЯ СИСТЕМ БІОМЕТРИЧНОЇ ІДЕНТИФІКАЦІЇ

1.1. Огляд сучасних методів біометричної ідентифікації

За сучасних умов розвитку суспільства проблема безпеки постає у новому аспекті – значна кількість об’єктів, яким потрібно забезпечити безпеку, існує у вигляді інформації, яка зберігається в електронних комп’ютерних системах та передається через мережі зв’язку. Необхідно забезпечити декілька рівнів захисту – обмежити фізичний доступ до електронних комп’ютерних систем (серверів), де зберігається інформація, забезпечити доступ до роботи з інформацією тільки акредитованим особам, забезпечити контроль фізичного доступу до приміщень, де знаходяться сервери і т. ін. Для підвищення ефективності систем захисту останнім часом пропонується використовувати так звані біометричні системи ідентифікації. Біометричні системи ідентифікації встановлюють особу за індивідуальними біометричними параметрами людини [1].

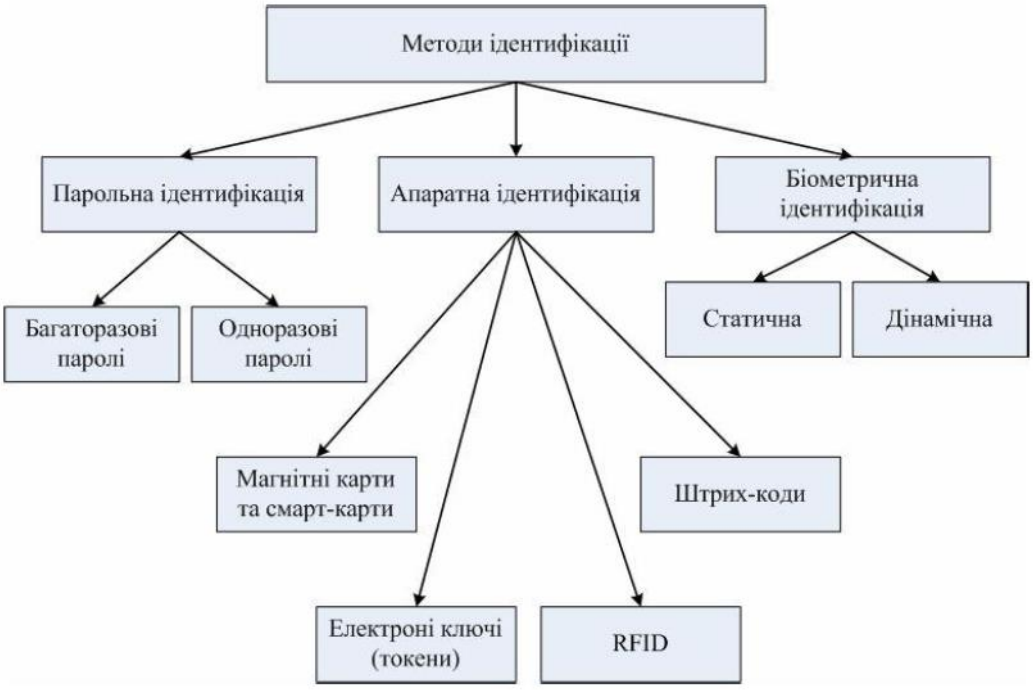


Рис. 1.1 – Методи ідентифікації

Біометричні ідентифікатори людини можна поділити на дві групи:

- фізіологічні (статичні) – засновані на фізіологічній (статичній) характеристиці людини, тобто унікальних властивостях, які властиві їй від народження і є невід'ємними від неї;

- психологічні (динамічні) – засновані на поведінковій (динамічній) характеристиці людини.

До найпоширеніших фізіологічних методів ідентифікації відносяться:

- Ідентифікація за відбитками пальців.
- Ідентифікація за сітчаткою ока.
- Ідентифікація за геометрією обличчя.
- Ідентифікація за розташуванням вен на лицьовій стороні долоні.
- Ідентифікація за райдужною оболонкою ока.
- Ідентифікація за допомогою ДНК.

Психологічних методів значно менше і до них відносяться:

- Ідентифікація за голосом.
- Ідентифікація за підписом (або почерком).
- Ідентифікація за клавіатурним почерком.

До найбільш розповсюджених методів біометричної ідентифікації відносяться способи, засновані на використанні наступних біометричних ідентифікаторів:

- відбитки пальців;
- райдужна оболонка ока;
- сітківка ока;
- геометрія обличчя;
- геометрія долоні;
- почерк (або підпис);
- ідентифікація за голосом.

Ці методи дозволяють ідентифікувати особу з досить високою ймовірністю. Крім цих біометричних ідентифікаторів сьогодні вже можна використовувати біометричну ідентифікацію на основі інших біометричних ідентифікаторів. До таких методів відносяться: ідентифікація на основі

термограми обличчя, термограми долоні та геометрії вуха. Крім того, виявлено біометричний ідентифікатор, який дозволяє отримувати стовідсотковий результат підчас проведення ідентифікації особи – це ідентифікація на основі ДНК людини. Проте цей метод має досить суттєвий недолік – він досить складний і вимагає значний обсяг часу [2].

Взагалі для розв'язання задачі ідентифікації особи у кожному з зазначених напрямів можна використовувати будь-який метод біометричної ідентифікації, проте, в залежності від конкретних умов задачі ідентифікації не завжди можна використовувати будь-який метод ідентифікації. Також значний вплив на популярність системи біометричної ідентифікації здійснює розвиток того чи іншого методу. Тобто метод може надавати ймовірність ідентифікації близьку до ста відсотків, але при цьому відсутні пристрої для його реалізації або чітко не визначена методика реалізації методу ідентифікації. Також на інтенсивність використання методу біометричної ідентифікації впливає його популярність. На сьогодні використовуються усі методи ідентифікації, проте деякі використовуються значно частіше ніж інші [2].

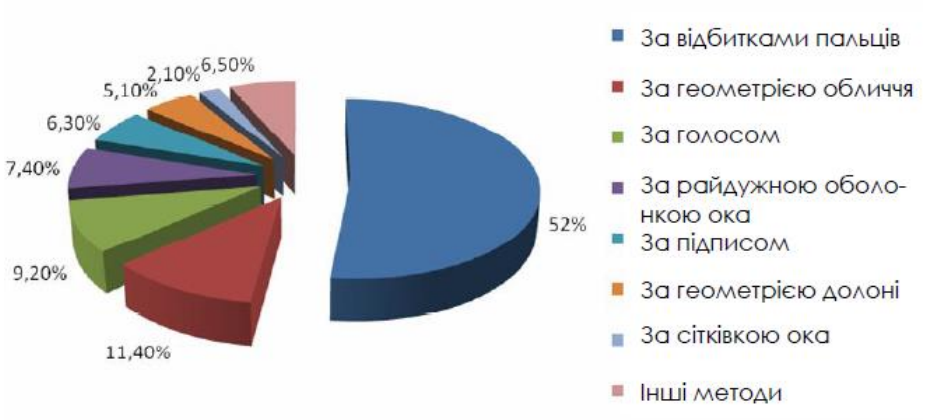
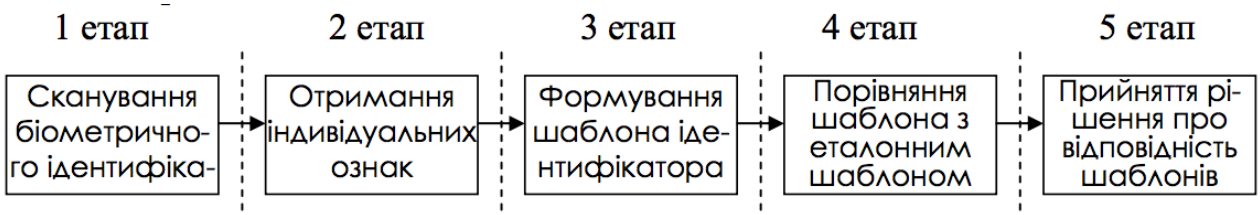


Рис. 1.2 - Діаграма використання методів біометричної ідентифікації

1.2. Принципи функціонування системи біометричної ідентифікації

Кожна система біометричної ідентифікації має свої особливості, які цілком залежать від типу біометричного ідентифікатора. Проте всі вони мають і деякі спільні риси, які притаманні усім системам. У загальному випадку системи біометричної ідентифікації працюють за наступним принципом. Усі

системи спочатку працюють у режимі реєстрації. В залежності від типу системи вона може використовувати декілька біометричних ідентифікаторів. Після отримання біометричного ідентифікатора система перетворює його за допомогою відповідних засобів в електронний вигляд. Ця стадія роботи системи біометричної ідентифікації називається реєстрацією, при цьому система отримує первісну інформацію, необхідну для її подальшої роботи. Звичайно система біометричної ідентифікації не зберігає зображення відбитка пальця, сітківки ока, геометрії долоні і т. ін. У системі зберігається так званий шаблон ідентифікатора, який являє собою одну або декілька цифрових послідовностей, які були отримані під час оброблення біометричного ідентифікатора. У будь-якому випадку незалежно від типу біометричного ідентифікатора, який застосовується системою, загальний алгоритм функціонування системи біометричної ідентифікації може бути наданий у



вигляді, показаному на рис. 1.3.

Рис. 1.3 – Загальний алгоритм функціонування систем біометричної ідентифікації

Слід зазначити, що процес ідентифікації у біометричних системах в цілому поділяється на два види – ідентифікацію та верифікацію. Різниця між цими двома поняттями надто тонка і досить часто один процес плутають з іншим. Ідентифікація – це порівняння типу “один-до-багатьох”, тобто здійснюється порівняння наданого біометричного ідентифікатора з усіма шаблонами біометричних ідентифікаторів, які є у базі. У результаті цього порівняння виявляється декілька найбільш схожих шаблонів (ті, які мають найбільшу вірогідність відповідності), а потім за допомогою будь-якого математичного критерію приймається рішення про найбільш ідентичний

шаблон. Верифікація – це порівняння типу “один-до-одного”, тобто здійснюється порівняння наданого ідентифікатора з відповідним шаблоном з бази. Однак в даному випадку необхідно надати додатковий ідентифікатор, який дозволить обрати з бази відповідний шаблон. Наприклад спочатку вводиться логін користувача, а потім надається відповідний біометричний ідентифікатор. У цьому режимі система ідентифікації працює набагато швидше та в повністю автоматичному режимі. Слід зазначити, що такий принцип роботи систем ідентифікації дозволяє розділити їх за сферами використання. Цілком зрозуміло, що система біометричної ідентифікації здатна працювати у двох режимах, проте більшість систем працює саме у режимі верифікації [3].

1.3. Фактори, які впливають на функціонування систем біометричної ідентифікації

На ефективність функціонування систем біометричної ідентифікації чинить вплив досить значна кількість факторів. Причому частина цих факторів визначається або властивостями самої системи біометричної ідентифікації, або процесом її функціонування, але існує також частина факторів, яка ніяк не залежить від системи біометричної ідентифікації, проте тісно з нею пов’язана та чинить значний вплив на ефективність роботи системи. У цілому до факторів, які впливають на ефективність біометричної системи відносяться [4]:

- властивості біометричного ідентифікатора;
- алгоритм ідентифікації;
- апаратна реалізація алгоритму;
- програмна реалізація алгоритму;
- ІТ-оточення, в якому система буде функціонувати;
- організаційна інфраструктура підприємства;
- соціальні та економічні обставини.

Отже, в даному розділі було виконано аналіз різних систем біометричної ідентифікації. Під час їх порівняння і вивчення стало зрозуміло які бувають системи біометричної ідентифікації, на чому базується принцип їх

функціонування. Ідентифікації за почерком, підписом, клавіатурним почерком і голосом є менш надійними з усіх. Мала надійність даних ідентифікаторів пов’язана не тільки з фізичними особливостями, але й з психологічними аспектами. Багатьма дослідженнями доведено, що голос, почерк, підпис людини досить сильно корелюється в залежності від психологічного стану людини. Ідентифікації за особливостями очей є досить точною, але має певні недоліки, такі як: не комфортний для людини процес ідентифікації, складність програмного і математичного забезпечення для проведення ідентифікації. Використовується в основному на об’єктах стратегічного значення з високим рівнем безпеки. Ідентифікації за особливостями долоні і вух – середня за складнощами реалізації і використання. Використовується в основному для ідентифікації особи, дані якої зафіксували камери відеоспостережень. Ідентифікація за обличчям достатньо розповсюджена. Даний метод ідентифікації достатньо вивчений і досліджений, це вирішує низку проблем з використанням даного методу, показує його надійність, точність і зручність. Ідентифікація за відбитками пальців – найдревніший і найбільш вивчений і досліджений метод ідентифікації. Простий у реалізації і використанні і в той же час досить надійний.

Всі вище описані методи можуть допускати помилки різного роду і не дають 100% точності ідентифікації. Але вони задовольняють висунутим умовам для того, щоб люди масово використовували дані методи. Єдиний відомий на сьогодні метод ідентифікації особи, який дає 100% результат – це ідентифікація за допомогою ДНК. Але в наш час даний метод не можливо використовувати у галузях, де ідентифікація особи має бути проведена за короткий час. Можливо з розвитком прогресу і технологій це стане можливим у майбутньому. З вище наведеного опису і порівняння для виконання БКР було обрано метод ідентифікації за відбитками пальців.

РОЗДІЛ 2. МЕТОДИ ПОРІВНЯННЯ ВІДБИТКІВ ПАЛЬЦІВ

2.1. Методика порівняння зображень відбитків пальців

Дактилоскопія - це встановлення особи людини за відбитками пальця, а точніше, по так званих папілярних узорах. Дактилоскопія ґрунтується на тому, що кожен відбиток пальця унікальний (за всю історію дактилоскопії не було виявлено двох співпадаючих відбитків пальців, що належать різним особам), а по-друге, папілярний візерунок не змінюється протягом усього життя людини. Шкіряний покрив пальців рук має складний рельєфний малюнок (папілярний візерунок), утворений лініями (висотою 0,1-0,4 мм і шириною 0,2-0,7 мм) і борозенками-заглибленнями (шириною 0,1-0,3 мм). Папілярний візерунок повністю формується на сьомому місяці розвитку плода людини. Більш того, в результаті проведених досліджень було встановлено, що відбитки пальців різні навіть у однояйцевих близнюків, хоча показники ДНК у них ідентичні. Крім того, папілярний узор неможливо видозмінити - ні порізи, ні опіки, ні інші механічні пошкодження шкіри не мають принципового значення, бо стійкість папілярного візерунка забезпечується регенеративною здатністю основного шару епідермісу шкіри. Тому можна стверджувати, що сьогодні дактилоскопія є найнадійнішим способом ідентифікації особистості [5].

Незважаючи на різноманіття будови папілярних візерунків, вони піддаються чіткій класифікації, що забезпечує процес їх індивідуалізації та ідентифікації. У кожному відбитку пальця можна визначити два типи ознак - глобальні та локальні. Глобальні ознаки - ті, які можна побачити неозброєним оком. Інший тип ознак - локальні. Їх називають мінуції - унікальні для кожного відбитку ознаки, що визначають пункти зміни структури папілярних ліній (закінчення, роздвоєння, розрив і т.д.), орієнтацію папілярних ліній і координати в цих пунктах. Практика показує, що відбитки пальців різних людей можуть мати однакові глобальні ознаки, але абсолютно неможлива наявність однакових мікровізерунків мінуцій. Тому глобальні ознаки

використовують для поділу бази даних на класи і на етапі аутентифікації. На другому етапі розпізнавання використовують вже локальні ознаки [10].

Принципи порівняння відбитків за локальними ознаками

Етапи порівняння двох відбитків.

1) Поліпшення якості вихідного зображення відбитка. Збільшується різкість кордонів папілярних ліній.

2) Обчислення поля орієнтації папілярних ліній відбитка. Зображення розбивається на квадратні блоки зі стороною більше 4 ПКС і по градієнтам яскравості обчислюється кут t орієнтації ліній для фрагмента відбитка.

3) Бінаризація зображення відбитка. Приведення до чорно-білого зображення (1 біт) пороговою обробкою.

4) Витончення ліній зображення відбитка. Потоншення проводиться до тих пір, поки лінії не будуть шириною 1 ПКС (рис. 2.1) [6].

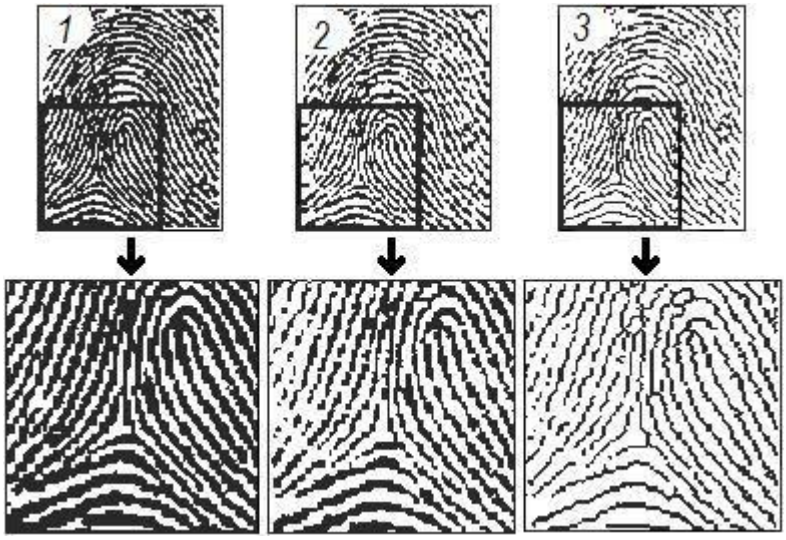


Рис. 2.1. Потоншення ліній зображення відбитка

5) Виділення мінуцій (рис. 2.2). Зображення розбивається на блоки 9 на 9 пікселів. Після цього підраховується число чорних (ненульових) пікселів, що знаходяться навколо центру. Піксель в центрі вважається мінуцією, якщо він сам ненульовий і сусідніх ненульових пікселів один (мінуція «закінчення») або два (мінуція «роздвоєння»).

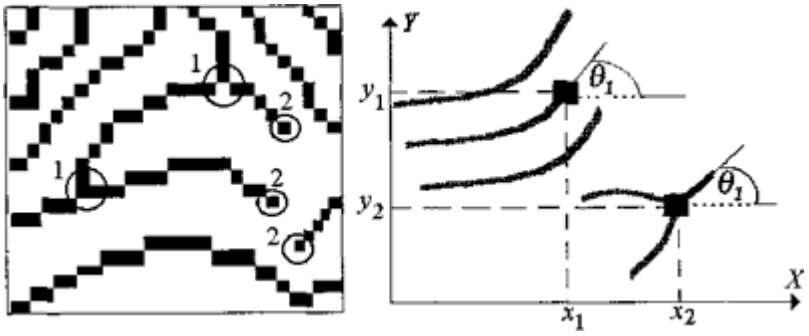


Рис. 2.2. Виділення мінуцій

Координати виявлених мінуцій і їх кути орієнтації записуються в вектор:

$$W(p) = [(x_1, y_1, t_1), (x_2, y_2, t_2)\ldots(x_p, y_p, t_p)], \quad (2.1)$$

де p – число мінуцій.

При реєстрації користувачів цей вектор вважається еталоном і записується в базу даних. При розпізнаванні вектор визначає поточний відбиток.

6) Зіставлення мінуцій. Два відбитка одного пальця будуть відрізнятися один від одного поворотом, зміщенням, зміною масштабу і / або площею дотику в залежності від того, як користувач прикладає палець до сканера. Тому не можна сказати, чи належить відбиток людині чи ні на підставі простого їх порівняння (вектори еталона і поточного відбитка можуть відрізнятися по довжині, містити невідповідні мінуції і т. д.). Через це процес зіставлення повинен бути реалізований для кожної мінуції окремо [11].

Етапи порівняння:

- реєстрація даних;
- пошук пар відповідних мінуцій;
- оцінка відповідності відбитків.

При реєстрації визначаються параметри афінних перетворень (кут повороту, масштаб і зміщення), при яких деяка мінуція з одного вектора є певною мінуцією з другого. При пошуку для кожної мінуції потрібно перебрати до 30 значень повороту (від -15° до + 15°), 500 значень зсуву (наприклад, від -250 ПКС до +250 ПКС) і 10 значень масштабу (від 0,5 до 1,5 з кроком 0,1). Разом до 150000 кроків для кожної з 70 можливих мінуцій. (На практиці всі

можливі варіанти не перебираються - після підбору потрібних значень для однієї мінуцій їх же намагаються підставити і до інших мінуцій, інакше було б можливо зіставити практично будь-які відбитки один одному) [3].

Оцінка відповідності відбитків виконується за формулою

$$K = (D \cdot D \cdot 100\ \%) / (p \cdot q), \qquad (2.2)$$

де D - кількість мінуцій, що зпівпали, p - кількість мінуцій еталона, q - кількість мінуцій ідентифікованого відбитка.

У разі, якщо результат перевищує 65%, відбитки вважаються ідентичними (поріг може бути знижений встановленням іншого рівня пильності). Якщо виконувалася аутентифікація, то на цьому все і закінчується. Для ідентифікації необхідно повторити цей процес для всіх відбитків в базі даних. Потім вибирається користувач, у якого найбільший рівень відповідності (зрозуміло, його результат повинен бути вище порога 65%).

2.2. Види і принцип роботи сканерів відбитків пальців

Пристроям для читання відбитків пальців в даний час знаходять широке застосування. Їх встановлюють на ноутбуки, в миші, клавіатури, флешки, а також застосовують у вигляді окремих зовнішніх пристроїв і терміналів, що продаються в комплекті з системами AFIS [7]. Незважаючи на зовнішні відмінності, всі сканери можна розділити на кілька видів:

1. Оптичні:
   - FTIR-сканери;
   - волоконні;
   - оптичні протяжні;
   - роликові;
   - безконтактні.
2. Напівпровідникові (напівпровідники змінюють властивості в місцях контакту):
   - ємнісні;
   - чутливі до тиску;

- термосканери;
- радіочастотні;
- протяжні термосканери;
- ємнісні протяжні;
- радіочастотні протяжні.

3. Ультразвукові (ультразвук повертається через різні проміжки часу, відбиваючись від борозенок або ліній).

Принцип роботи сканера відбитків пальців, як і будь-якого іншого пристрою біометричної верифікації, досить простий і включає чотири базових етапи:

- запис (сканування) біометричних характеристик (в даному випадку - пальців);
- виділення деталей папілярного візерунка по декількох точках;
- перетворення записаних характеристик в відповідну форму;
- порівняння записаних біометричних характеристик з шаблоном;
- прийняття рішення про збіг або розбіжності записаного біометричного зразка з шаблоном.

Ємнісні сенсори складаються з масиву конденсаторів, кожен з яких представляє собою дві з'єднані пластини. Ємність конденсатора залежить від прикладеної напруги і від діелектричної проникності середовища. Коли до такого масиву конденсаторів підносять палець, то і діелектрична проникність середовища, і ємність кожного конденсатора залежать від конфігурації папілярного візерунка в локальній точці. Таким чином, по ємності кожного конденсатора в масиві можна однозначно ідентифікувати папілярний візерунок.

Принцип дії оптичних сенсорів подібний до того, що використовується в побутових сканерах. Такі сенсори складаються з світлодіодів і ПЗС-сенсорів: світлодіоди висвітлюють скановану поверхню, а світло, відбиваючись, фокусується на ПЗС-сенсори. Оскільки коефіцієнт відбиття світла залежить від будови папілярного візерунка в конкретній точці, то оптичні сенсори дозволяють записувати образ відбитку пальця [8].

Термічні сенсори являють собою масив піроелектриків - це різновид діелектриків, на поверхні яких при зміні температури виникають електричні заряди через зміну спонтанної поляризації. Температура в міжпапілярних западинах нижче, ніж на поверхні валика папілярної лінії, внаслідок чого масив піроелектриків дозволяє в точності відтворити папілярний візерунок.

У сенсорах електромагнітного поля є генератори змінного електричного поля радіочастоти і масив приймальних антен. Коли до сенсора підносять палець, то генеруються силові лінії електромагнітного поля, які в точності повторюють контур папілярних ліній, що дозволяє масиву приймальних антен фіксувати структуру відбитка пальця.

Розглянемо більш докладно принцип роботи протяжних термосканерів - найпопулярніших в наш час. У них реалізований тепловий метод зчитування відбитків пальців, заснований на властивості піроелектричних матеріалів перетворювати різницю температур в напругу.

Різниця температур створюється між осередками чутливого елемента під папілярними гребінцями і боріздками. Боріздки не контактують з чутливим елементом, тому температура чутливого елемента під боріздками залишається рівною температурі навколишнього середовища. Особливістю температурного методу є те, що через деякий час (близько 0,1 с) зображення зникає, оскільки палець і датчик приходять в температурну рівновагу.

Швидке зникнення температурної картини є однією з причин застосування технології сканування. Щоб отримати відбиток, потрібно провести пальцем поперек чутливого елемента прямокутної форми (0,4 на 14 мм або 0,4 на 11,6 мм). Під час руху пальця швидкість сканування повинна перевищувати 500 кадрів / с (задається тактовою частотою). В результаті виходить послідовність кадрів, кожен з яких містить частину загальної картини. Далі відбиток пальця реконструюють програмним способом: в кожному кадрі вибирають кілька ліній пікселів і шукають ідентичні лінії в інших кадрах, повний образ відбитку пальця отримують суміщенням кадрів на основі цих ліній.

Метод покадрового зчитування не вимагає розрахунку швидкості руху пальця по зчитувачу і дозволяє зменшити площу кремнієвої підкладки матриці більш ніж в 5 разів, що в стільки ж разів знижує її вартість.

Стандартизація шаблонів відбитків пальців

Зараз в основному використовуються стандарти ANSI і ФБР США. У них визначені наступні вимоги до образу відбитка [8]:

- кожен образ представляється у форматі нестислого TIF;
- образ повинен мати роздільну здатність не нижче 500 dpi;
- образ повинен бути півтоновим з 256 рівнями яскравості;
- максимальний кут повороту відбитка від вертикалі не більше 15°;
- основні типи мінуцій - закінчення і роздвоєння.

Зазвичай в базі даних зберігають більше одного способу, що дозволяє поліпшити якість розпізнавання [9]. Образи можуть відрізнятися один від одного зміщенням і поворотом. Масштаб не змінюється, так як всі відбитки отримують з одного пристрою.

Отже, в даному розділі наведено методи порівнянь відбитків пальців. Описаний метод порівнянь за локальними ознаками, який включає наступні етапи: поліпшення якості вихідного зображення відбитка, обчислення поля орієнтації папілярних ліній відбитка, бінаризація зображення відбитка, витончення ліній зображення відбитка, виділення мінуцій, зіставлення мінуцій.

Описано різновиди сканерів відбитків пальців, а також приведено опис їх функціонування. Оскільки аутентифікаційні системи зберігають дуже значущі і важливі дані (біометричні дані користувачів) питання безпеки є вкрай важливим і ним не можна нехтувати при розробці подібних систем.

РОЗДІЛ 3. РОЗРОБКА СИСТЕМИ КОНТРОЛЮ ДОСТУПУ НА ОСНОВІ БІОМЕТРИЧНОГО ІДЕНТИФІКАТОРА

В даному розділі наведено результати розробки системи ідентифікації особи за відбитками пальців з використанням сканера відбитків пальців і плати Arduino Uno. Описана технічна і програмна складові проектованої системи.

3.1. Апаратна реалізація проекту

Плата Arduino Uno

Плата Arduino Uno - це пристрій на основі мікроконтролера ATmega328. У його склад входить все необхідне для зручної роботи з мікроконтролером: 14 цифрових входів / виходів (з них 6 можуть використовуватися в якості ШІМ-виходів), 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування (ICSP) і кнопка скидання. Для початку роботи з пристроєм досить просто подати живлення від AC/DC-адаптера або батарейки, або підключити його до комп'ютера за допомогою USB-кабелю. На відміну від всіх попередніх плат Arduino, Uno в якості перетворювача інтерфейсів USB-UART використовує мікроконтролер ATmega16U2 (ATmega8U2 до версії R2) замість мікросхеми FTDI. На платі Arduino Uno версії R2 для спрощення процесу оновлення прошивки доданий резистор, що підтягує до землі лінію HWB мікроконтролера 8U2. Зміни на платі версії R3 перераховані нижче: додані терморегулятори 1.0, виводи SDA і SCL (біля AREF), а також два нових виводи, розташованих біля виводу RESET. Перший - IOREF - дозволяє платам розширення підлаштовуватися під робочу напругу Arduino. Даний вивід передбачений для сумісності плат розширення як з 5В- Arduino на базі мікроконтролерів AVR, так і з 3.3В-платами Arduino Due. Другий вивід ні до чого не приєднаний і зарезервований для майбутніх цілей. Покращена стійкість ланцюга скидання. Мікроконтролер ATmega8U2 замінений на ATmega16U2. "Uno" (в перекладі з італійської - "один") названий з нагоди майбутнього випуску Arduino 1.0.

Спільно з Arduino 1.0 дані пристрої будуть базовими версіями Arduino. Uno - еталонна модель платформи Arduino і є останньою в серії USB-плат.

Характеристики плати Arduino Uno наведені в таблиці 3.1.

Таблиця 3.1

Мікроконтролер	ATmega328
Робоча напруга	5В
Напруга живлення (рекомендована)	7-12В
Напруга живлення (гранична)	6-20В
Цифрові входи/виходи	14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
Аналогові входи	6
Максимальний струм одного виводу	40 мА
Максимальний вихідний струм виводу	3.3V 50 мА
Flash-пам'ять	32 КБ (ATmega328) з яких 0.5 КБ використовуються завантажувачем
SRAM	2 КБ (ATmega328)
EEPROM	1 КБ (ATmega328)
Тактова частота	16 МГц

Рис. 3.1 – Плата Arduino Uno

Сканер відбитка пальця FPM10A

Оптичний сканер відбитків пальців на процесорі ARM Cortex M 32-bit - Synochip AS608 (FPM10A) підтримує алгоритми шифрування даних. Створює базу відбитків у внутрішній пам'яті, виконує порівняння за шаблоном. Налаштування проводиться за допомогою утиліти від виробника або скетчів для Arduino.

Технічні характеристики даного сканеру наведені в таблиці 3.2.

Таблиця 3.2

Напруга живлення	3.6 - 6.0 В
Струм	120 мА (140 мА макс.)
Час обробки зображення відбитка	<1.0 секунди
Сенсор	оптичний
Розмір сенсора	14 мм х 18 мм
Розмір сигнатури	256 байт
Розмір шаблону	512 байт
Ємність	300 осередків
Рівні безпеки	(1-5)
Інтерфейс	UART TTL
Швидкість передачі даних	9600, 19200, 28800, 38400, 57600 (за замовчуванням 57600)
Робоча температура	-20 С - +50 С
Допустимий рівень вологості	40% - 85% RH
Габаритні розміри	45 х 26 х 19 мм
Вага	15 гр.

Рис. 3.2 - Сканер відбитка пальця FPM10A

LCD дисплей 2004 I2C

LCD дисплей має 20 символів в кожному з чотирьох рядків з керуванням по шині I2C (TWI, IIC), що дуже зручно при нестачі вільних виводів на Arduino, досить підключити всього два дроти (без врахування живлення) і можна повністю керувати дисплеєм.

Технічні характеристики LCD дисплею наведені в таблиці 3.3

Таблиця 3.3

I2C адреса	0x27 або 0x3f (в залежності від версії I2C адаптера)
Чіп конвертера	I2C: PCF8574A
Кількість символів в рядку	20
Кількість рядків	4
Підтримка кирилиці	відсутня
Колір фону Колір символів	синій білий

Продовження таблиці 3.3

Контраст	налаштовується резистором зі зворотного боку плати
Живлення	5 В
Розміри точки	0.55 х 0.55 мм
Крок точки	0.60 х 0.60 мм
Розміри символу	2.96 х 4.75 мм
Крок символів	3.55 х 5.35 мм
Розміри	98 х 60 х 20 мм

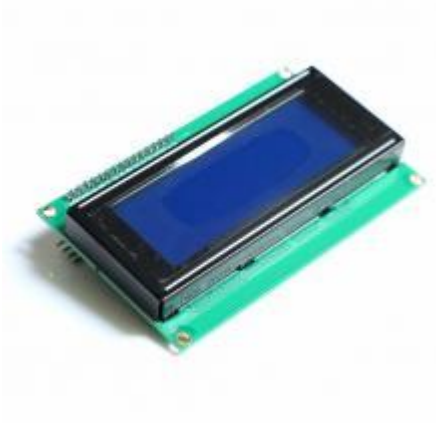


Рис. 3.3 - LCD дисплей 2004 I2C

Плата Arduino Uno в проекті виконує свою логічну функцію, а саме: зв’язує всі компоненти і керує ними. Також в пам’яті плати зберігається програмний код всього проекту. Завдяки цьому система є повністю автономною, за наявності джерела живлення. Сканер відбитка пальця має свою власну мікросхему і пам'ять, в якій зберігається база відбитків пальців. Максимально пам'ять сканеру може одночасно зберігати 127 відбитків. Сканер керується своїм власним програмним забезпеченням, яке міститься в його мікросхемі. Для зв’язування сканеру з платою Arduino достатньо встановити необхідні бібліотеки, щоб мати можливість написання програмного коду. LCD дисплей дозволяє виводити меню програми і всю інформацію, необхідну користувачеві. Управління виконується за допомогою чотирьох кнопок які виконують відповідно наступні функції «Return», «Up», «Down», «OK». Два

світлодіода показують інформацію стану виконання операцій. Зелений – операція пройшла успішно, і червоний, відповідно, інформує про помилки. Макет проекту наведено в додатку А, принципова схема наведена у додатку Б.

3.2. Програмна реалізація проекту

Програмне забезпечення розроблено в середовищі Arduino IDE на мові програмування C++. Для функціонування всіх модулів необхідно встановити додаткові бібліотеки. Бібліотеки, які використанні у проекті наведено нижче:

- Adafruit_Fingerprint.h – необхідна для роботи зі сканером відбитків пальців;
- Wire.h - дана бібліотека дозволяє взаємодіяти з I2C / TWI пристроям;
- LiquidCrystal_I2C.h – бібліотека для роботи з LCD дисплеєм.

На початку проходить ініціація портів і виводів для сканеру відбитків пальців і LCD дисплею. Оголошуються константні і загальні змінні. Далі слідує функція void setup() – в ній описаний код, який виконується лише один раз на початку програми. В даній функції запускається дисплей і обнуляються його дані виводу, це необхідно для того, щоб на екран не виводилась інформація з попередньої сесії запуску програми. Також задається швидкість передачі інформації для LCD дисплею і сканера відбитків пальців. Перевіряється чи підключений сканер і чи немає проблем зв’язку між платою Arduino і сканером відбитків пальців. Якщо на даному етапі є проблеми, на дисплей виводиться відповідне повідомлення помилки (рис 3.4):

Рис. 3.4 – Зображення LCD дисплею з повідомленням помилки

Якщо ніяких проблем не виникло, на екран виводиться відповідне повідомлення (рис. 3.5):

Рис 3.5 – Зображення LCD дисплею на початку роботи

Також в момент запуску і ініціації на дисплей виводиться інформація про кількість відбитків в базі даних. База даних відбитків зберігається у внутрішній пам’яті сенсора відбитків пальців. Об’єм пам’яті дозволяє одночасно зберігати 127 відбитків, але в даній роботі програмно задане обмеження у розмірі 20 відбитків.

Система працює в наступних режимах роботи:

- режим «Сканування», «Scan mode»;
- режим «Додавання нового відбитку», «Add new fingerprint»;
- режим «Видалення відбитку», «Delete fingerprint».

Також в програмі передбачений пункт «Інформація» («Info»), який показує скільки відбитків зберігається в базі даних.

Режим роботи системи «Сканування»

Після того як ініціація всіх пристроїв пройшла успішно, система автоматично переходить в режим «Сканування». На дисплеї виводиться відповідне повідомлення (рис. 3.6):

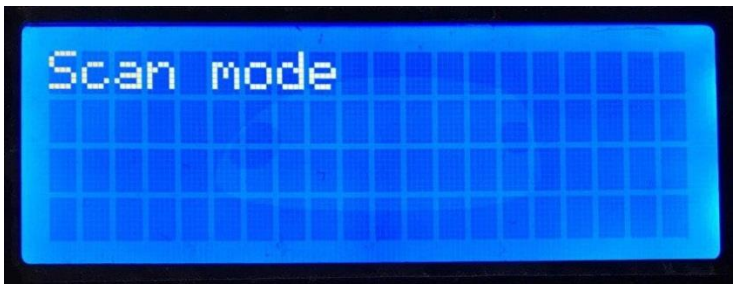


Рис 3.6 – Система в режимі «Сканування»

В даному режимі роботи система завжди знаходиться в стані очікування. Проводиться постійна перевірка стану сканера відбитків пальців, якщо до

сканеру прикладено палець – система одразу ідентифікує його. Якщо даний відбиток зберігається в базі даних, система проінформує користувача про успішну ідентифікацію, загориться зелений світлодіод, на дисплеї з'явиться відповідне повідомлення (рис 3.7).

Рис 3.7 – Повідомлення успішної ідентифікації.

Також на дисплей виводиться ID (від англ. «Identifier» - ідентифікатор) і відповідний номер. За цим номером в базі даних зберігається щойно ідентифікований відбиток пальця. Тобто система розпізнала палець, прикладений до сканеру, як шаблон, який зберігається в базі даних. Якщо ж прикладений палець не відповідає ні одному з шаблонів в базі даних, на дисплеї відобразиться відповідне повідомлення і загориться червоний світлодіод (рис. 3.8).

Рис. 3.8 – Повідомлення помилки ідентифікації.

Після проведення успішної або не успішної ідентифікації система в режимі «Сканування» знову переходить в стан очікування.

Режим роботи системи «Додавання нового відбитку в базу даних»

Перемикання, тобто пролистування пунктів меню, які є відображенням режимів роботи, здійснюється за допомогою кнопок «Up» («Верх») і «Down»

(«Вниз»). Для того, щоб перемкнути систему в обраний режим роботи, слід натиснути кнопку «ОК». Щоб вийти з режиму роботи і мати змогу обрати інший, треба натиснути кнопку «Return» («Повернення»). Після того, як був обраний режим «Додавання нового відбитку», система запропонує користувачеві обрати номер ID для нового відбитку. Відображаються лише доступні ID (рис. 3.9).

Рис. 3.9 – Режим «Додавання нового відбитку»

Після того, як ID обраний, почнеться процедура зняття нового відбитка. Спочатку треба прикласти палець до сканеру, після зняття першого відбитку відобразиться повідомлення про те, що шаблон був знятий і конвертований. Після цього треба забрати палець зі сканеру і прикласти той самий палець другий раз. Якщо палець прикладено неправильно (під викривленим кутом або не повністю), система буде знаходитись в режимі очікування допоки не буде знятий другий шаблон відбитку пальця. Після цього на дисплеї відобразиться повідомлення про те, що шаблон був знятий, конвертований, перший і другий шаблони збігаються, відбиток збережено в базі (рис 3.10).

Рис. 3.10 – Успішне зняття і збереження відбитка в БД

Повідомлення про конвертацію в даному випадку – це перетворення зображення в цифровий код координат і даних, який потім зберігається в базі даних. Саме за цим кодом потім здійснюється ідентифікація відбитків.

Пункти «Видалення відбитків з БД» і «Інформація»

Щоб видалити відбиток з бази даних необхідно обрати режим роботи (пункт меню) «Delete fingerprint», натиснути кнопку «ОК» і після цього обрати конкретний ID відбитка для видалення. Після цього ще раз натиснути кнопку «ОК». Після цього відбиток буде видалений. При пролистуванні номерів ID показуються лише ті ID під якими зберігаються відбитки в базі даних. «Порожні» ID не показуються. Також є можливість видалили всі відбитки одразу, очистивши при цьому базу даних (рис 3.11).

Рис 3.11 – Видалення відбитка з БД

Пункт «Info» («Інформація») – виводить на дисплей повідомлення, в якому вказана кількість відбитків пальців, які зберігаються в базі даних в поточний момент. Блок-схема алгоритму програми наведена в додатку В. Код програми наведено в додатку Д.

3.3. Аналіз роботи системи

Провівши низку випробувань було опрацьовано і оцінено роботу системи, її точність, ефективність. Було проведено 50 експериментів ідентифікації одного і того ж самого відбитка пальця для встановлення низки параметрів. Таблиця, в якій наведені дані даного експерименту, наведена в додатку Г. Ступінь достовірності (Confidence level) – це число, яке показує на скільки відбиток ідентифікованого пальця співпадає з відбитком, який зберігається в базі даних пристрою. У середовищі розробки Arduino IDE бібліотека для

роботи зі сканером відбитка пальця «Adafruit_Fingerprint.h» містить функцію finger.confidence, яка в свою чергу показує ступінь достовірності в умовних одиницях. В даній роботі вивід значень даної функції відбувався на монітор COM порта. На рис. 3.12 показані дані експерименту. По вертикалі відображені значення ступеню достовірності, по горизонталі – номери дослідів.

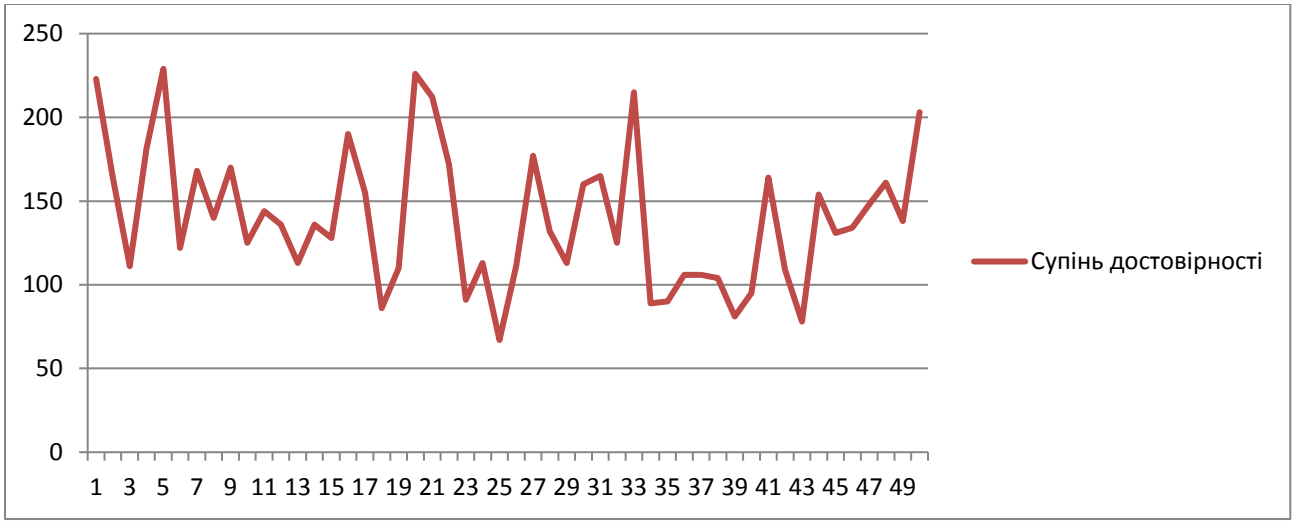


Рис. 3.12 – Ступінь достовірності при всіх позитивних ідентифікаціях

- Максимальне значення ступеню достовірності - 229
- Мінімальне значення ступеню достовірності – 67

Ступінь достовірності встановлює порогове значення ідентифікації, тобто на скільки має співпасти відбиток прикладеного пальця з інформацією про нього, яка зберігається в базі даних, щоб ідентифікація пройшла успішно. Регулюючи рівень порогового значення коригують вірогідності помилки першого та другого роду FFR і FAR (див. рис. 1.7).

- При встановленні порогового значення ступеню достовірності більше 120 у.од. – 64% від всіх ідентифікацій мали би успіх і 36% ідентифікацій було би відхилено.
- При встановленні порогового значення ступеню достовірності більше 110 у.од. – 74% від всіх ідентифікацій мали би успіх і 26% ідентифікацій було би відхилено.

- При встановленні порогового значення ступеню достовірності більше 100 у.од. – 84% від всіх ідентифікацій мали би успіх і 16% ідентифікацій було би відхилено (рис. 3.13).

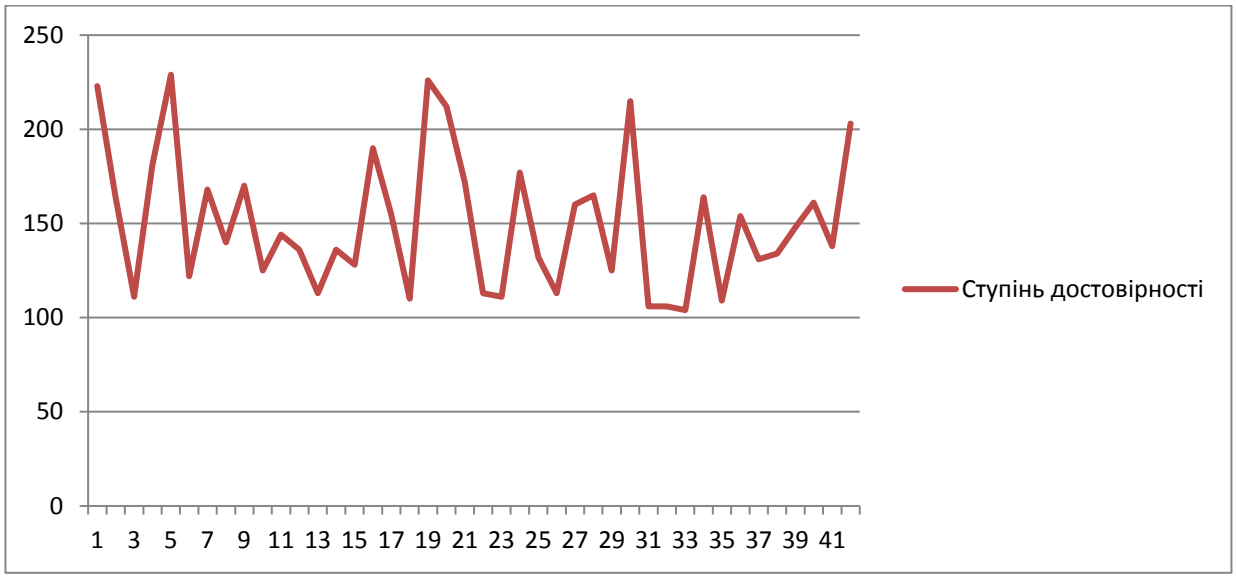


Рис. 3.13 - Ступінь достовірності при всіх позитивних ідентифікаціях при встановленні порогового значення більше 100

Під час проведення експерименту не було виявлено ні одного випадку помилки першого і/або другого роду. Помилки першого роду виникали лише при навмисному неправильному прикладанні пальця до сканеру (під викривленими кутами, не повністю поклавши палець на сканер, слабо або сильно натискаючи пальцем на сканер).

Отже, в даному розділі описано етапи проектування і створення системи біометричної ідентифікації користувачів. Приведено опис, технічні характеристики і переваги плати Arduino Uno, сканеру відбитків пальців FPM10A і LCD дисплею 2004 I2C – як основних технічних складових проекту. Приведений опис програмного забезпечення. Описано внутрішню логічну будову програмного забезпечення. Приведений детальний опис основних режимів роботи системи. Режим роботи «Сканування» - виконує безпосередньо функцію ідентифікації користувача. Якщо відбиток пальця даного користувача був зареєстрований і зберігається в базі даних, система ідентифікує його, на дисплей буде виведено ID ідентифікованого користувача. Зелений світлодіод проінформує про успішне проходження процедури ідентифікації. Якщо ж

система не ідентифікує користувача – буде виведено відповідне повідомлення на дисплей, і загориться червоний світлодіод помилки. Режим роботи «Додавання відбитку в базу даних» реалізовує реєстрацію користувачів. Реєстрація займає в середньому 15-30 секунд, що є досить гарним показником швидкості роботи системи. Пункт «Видалення відбитків з бази даних» розроблений з урахуванням можливості видалення окремих відбитків з бази даних і можливості видалити всю базу даних одразу.

Було проведено експеримент, опрацювання результатів якого показало:

- відсутність помилок першого роду;
- відсутність помилок другого роду;
- достатня швидкість роботи системи.

Також був обрахований оптимальний поріг ідентифікації – більше 100 у.од.

В додатку А приведено макет проекту, в додатку Б – принципову схему, в додатку В – блок-схему алгоритму програмного забезпечення, в додатку Г – таблицю результатів проведеного експерименту, в додатку Д – код програмного забезпечення.

ВИСНОВКИ

В роботі було проведено аналіз існуючих систем біометричної ідентифікації. Досліджено основні біометричні ідентифікатори людини, зручності і недоліки використання тих чи інших даних людини в якості біометричних ідентифікаторів. В результаті аналізу, для побудови системи ідентифікації в якості біометричного ідентифікатора був обрано зображення відбитка пальця людини. Наведено методи і принципи порівнянь відбитків пальців. Було проведено аналіз різних сканерів відбитків пальців, принцип і технології на яких базується їх функціонування. Для розробки системи біометричної ідентифікації було обрано сканер відбитків пальців FPM10A.

В роботі було розроблено систему біометричної ідентифікації за відбитками пальців. Було розроблено супроводжуюче програмне забезпечення, яке включає в себе декілька режимів роботи. Приведено експеримент для вивчення і аналізу роботи системи. Проектована система в ході експерименту не показала помилок першого і помилок другого роду. Швидкодія системи складає до 30 секунд в режимі реєстрації нового користувача та 5 секунд в режимі аутентифікації.

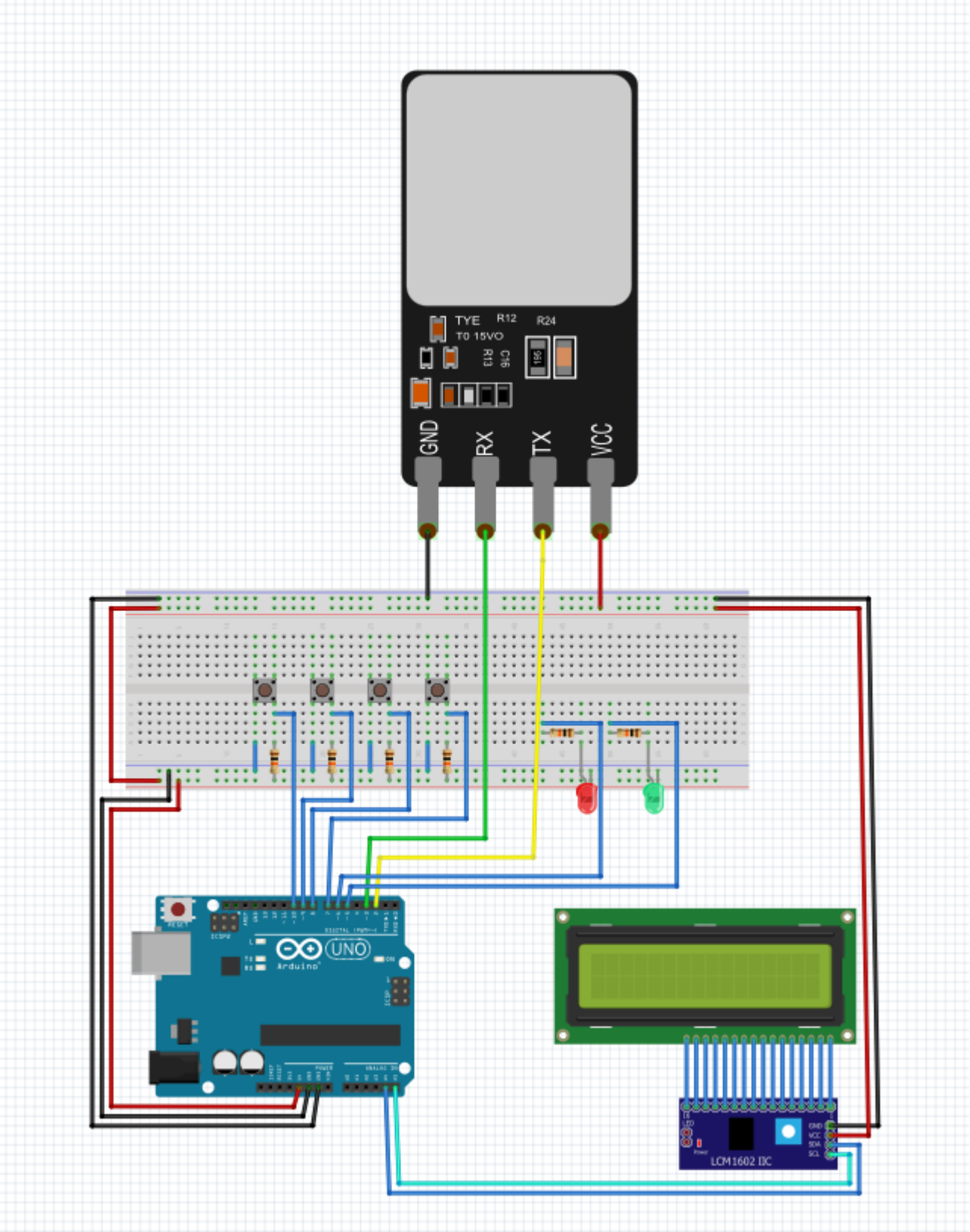
В результаті тестування системи виявлено, що при встановленні порогового значення ступеню достовірності більше 120 умовних одиниць кількість успішних ідентифікацій складає 64%, а при 100 у.од. – 84% від усіх спроб ідентифікації. Виявлено, що при неправильному прикладанні пальця до сканеру система не може ідентифікувати користувача. Максимальна кількість відбитків, що може зберігатись у базі даних складає 127.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. https://en.wikipedia.org/wiki/Biometrics [Електронний ресурс].
2. Царьов Р. Ю. Біометричні технології: навч. посіб. [для вищих навчальних Ц18 закладів] / Р. Ю. Царьов, Т. М. Лемеха. – Одеса: ОНАЗ ім. О. С. Попова, 2016. – 140 с.: іл.
3. Прудник А. М. Биометрические методы защиты информации : учеб. – метод. пособие / А. М. Прудник, Г. А. Власова, Я. В. Рощупкин. – Минск : БГУИР, 2014. – 123 с. : ил.
4. Захаров В. П., Рудешко В. І. З-38 Біометричні технології в XXI столітті та їх використання правоохоронними органами: посібник. – 2-ге вид., доп. / В. П. Захаров, В. І. Рудешко. – Львів: ЛьвДУВС, 2015. – 492 с.
5. Anil K. Jain, Arun A. Ross, Karthik Nandakumar. Introduction to Biometrics. Foreword by James Wayman.
6. Болл Р. М. Руководство по биометрии / Р.М. Болл [и др.]. – М., 2007.
7. Нгуен Вьет Хунг. Методы биометрической идентификации личности по внешним статическим образам – Общая структура систем / Нгуен Вьет Хунг // Курс "Защита информации", кафедра радиотехники МФТИ .
8. Biometrics catalog [Електронний ресурс]. – Режим доступа: http://www.biometricscatalog.org/, англ.
9. Задорожный В. Обзор биометрических технологий // Защита информации. Конфідент. – 2003. – № 5. – С. 19-25.
10. B. Miller. Vital signs of identity. IEEE Spectrum, 31 (2) : 22–30,1994.
11. R. M. Bolle, J. H. Connell, S. Pankanti and N. Ratha. On the security of biometrics authentication. IBM Technical Report, 2002.

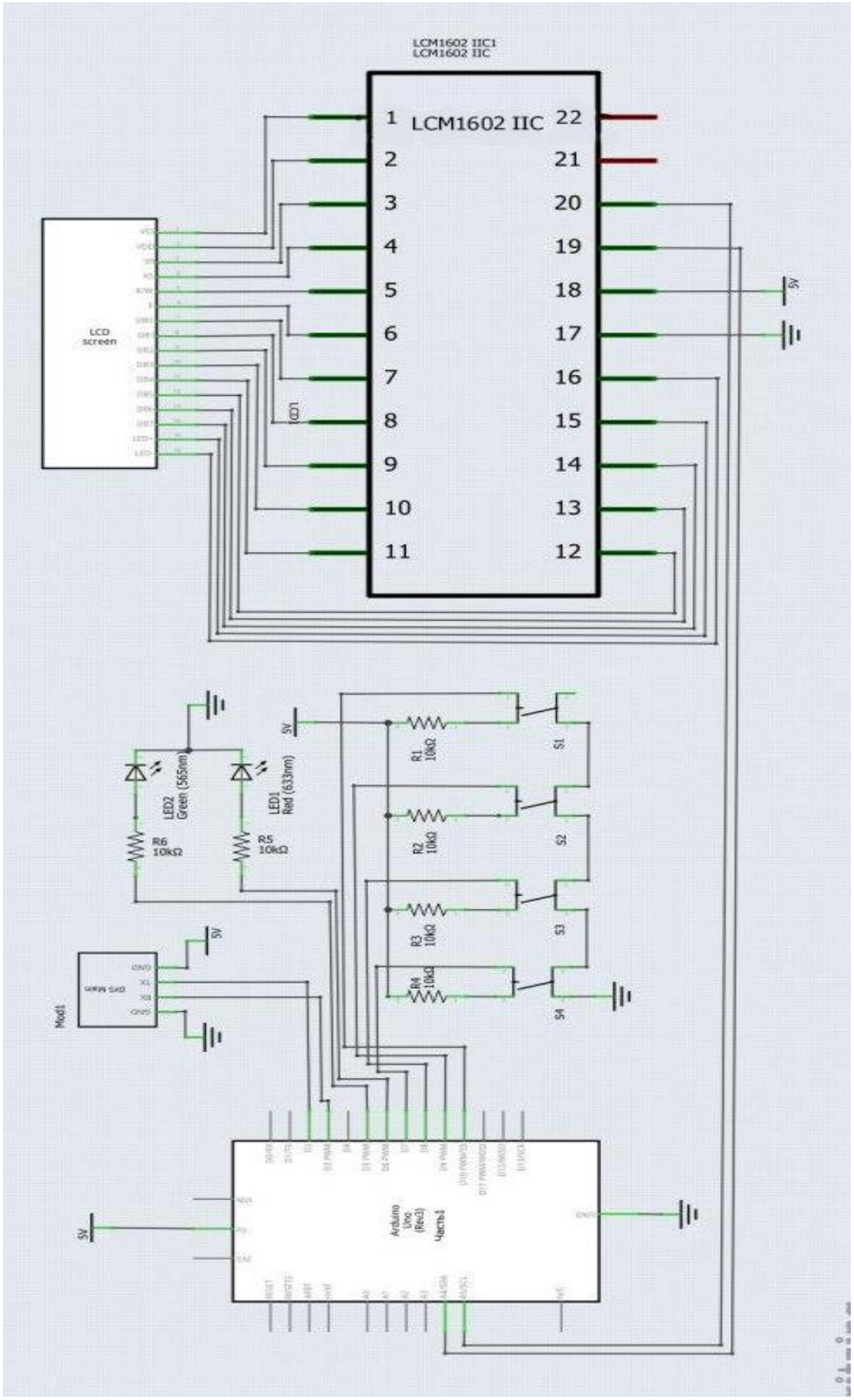
Додаток А

Макет проекту



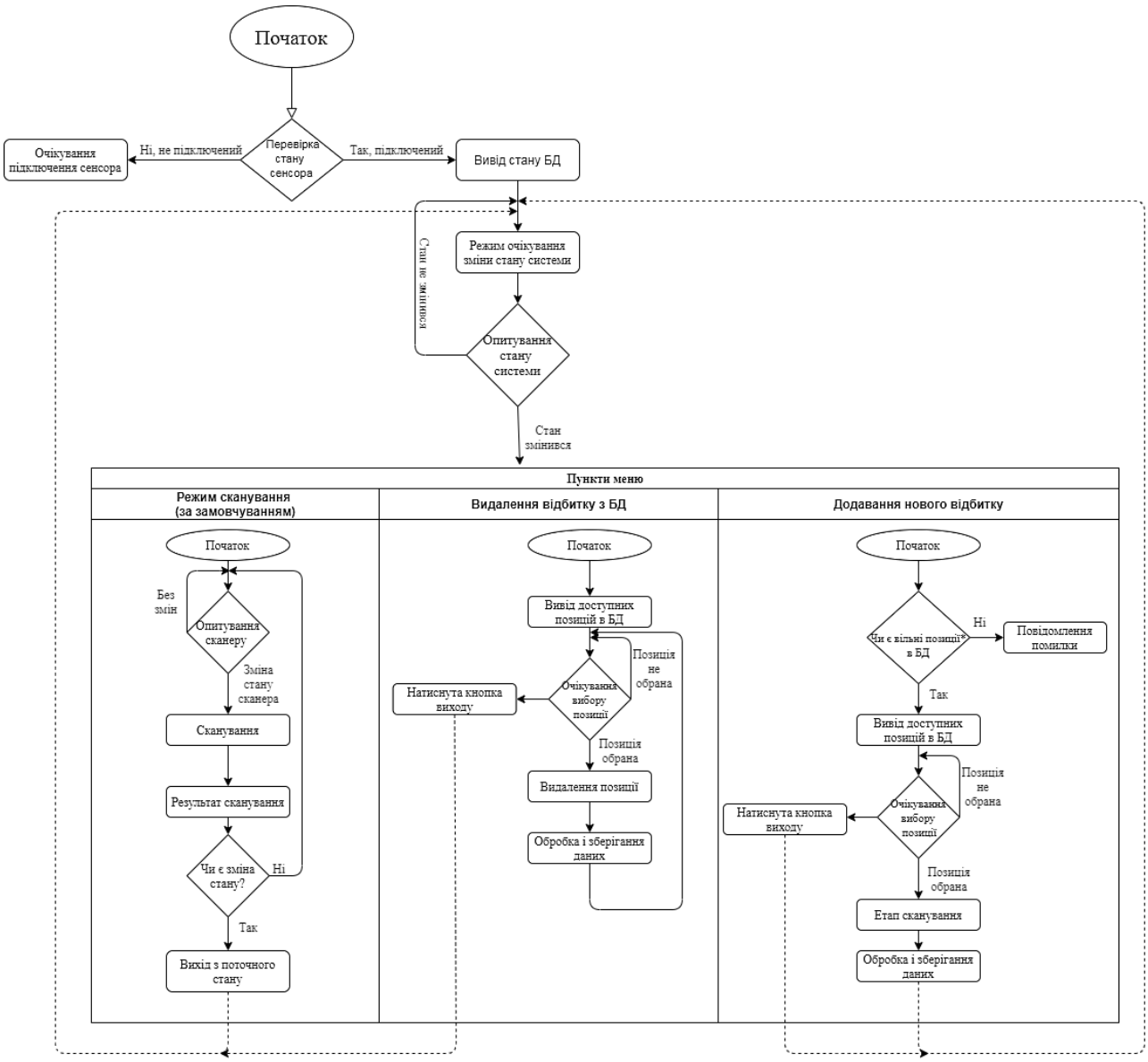
Додаток Б

Принципова схема



Додаток В

Блок-схема програмного забезпечення



*

під "позицією" мається на увазі намер ID відбитка у базі

Додаток Г

Таблиця дослідів ступеню достовірності

№ досліду	Ступінь достовірності (у.од.)	№ досліду	Ступінь достовірності (у.од)
1	223	26	111
2	164	27	177
3	111	28	132
4	181	29	113
5	229	30	160
6	122	31	165
7	168	32	125
8	140	33	215
9	170	34	89
10	125	35	90
11	144	36	106
12	136	37	106
13	113	38	104
14	136	39	81
15	128	40	95
16	190	41	164
17	155	42	109
18	86	43	78
19	110	44	154
20	226	45	131
21	212	46	134
22	172	47	148
23	91	48	161

Продовження таблиці дослідів ступеню достовірності

№ досліду	Ступінь достовірності (у.од.)	№ досліду	Ступінь достовірності (у.од.)
24	113	49	138
25	67	50	203

Додаток Д

Код програмного забезпечення

```
#include <Adafruit_Fingerprint.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
SoftwareSerial mySerial(2, 3);
LiquidCrystal_I2C lcd(0x27,20,4);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
const int buttonUpPin = 10;
const int buttonDownPin = 9;
const int buttonOkPin = 8;
const int buttonReturnPin = 7;
const int PIN_OPEN_OK = 6;
const int PIN_OPEN_ERROR = 5;
// маскимальний ID
const int MaxID = 20;
int curentMenu = 0 ;
int outMenu = 100;
int curentID = 1 ;
char *menuString[] = {
  "Scan mode",
  "Add new fingerprint",
  "Delete fingerprint",
  "Info" };
// максимальний пункт меню дорівнює кількості стрічок в *menuString - 1
int MaxMenuItem = 3;
void setup() { lcd.init(); lcd.backlight();  lcd.clear();  lcd.home();
 finger.begin(57600);
  pinMode(buttonUpPin, INPUT);
  pinMode(buttonDownPin, INPUT);
  pinMode(buttonOkPin, INPUT);
  pinMode(buttonReturnPin, INPUT);
  pinMode(PIN_OPEN_ERROR, OUTPUT);
  pinMode(PIN_OPEN_OK, OUTPUT);
  Serial.begin (9600);
  // Serial.println(MaxMenuItem);
  if (finger.verifyPassword()) {
    outString("Sensor is found!", 0,0,1);
    finger.getTemplateCount();
    outString(String(finger.templateCount,10)+ " fingerprints", 0,1,0);
    outString("are stored", 0,2,0);
    outString("in a database", 0,3,0);  }
  else {
```

```
    outString("A fingerprint sensor", 0,0,1);
    outString("was not found", 0,1,0);
    while (1) { delay(1); } }
  delay(5000); }
void loop() {  int p = -1;
// Коли приклали палець на сканер
p = finger.getImage();
if( p == FINGERPRINT_OK ) { scanFingerPrint(); }
 menuControl() ;
 runMenuItem() ;
 if( outMenu != curentMenu ) { outString(menuString[curentMenu], 0,0, 1);
  outMenu = curentMenu ; } delay(10); }
void scanFingerPrint() {
  int p = -1;  p = finger.image2Tz();
  if( p == FINGERPRINT_OK ) { p = finger.fingerFastSearch();
//Якщо палець є в БД, вмикається зелений світлодіод
      if( p == FINGERPRINT_OK ) {
        digitalWrite( PIN_OPEN_OK, HIGH);
        outString( "Found fingerprint" ,0,0,1);
        outString( "ID = " + String(finger.fingerID,10),0,1,0); delay(5000);
        digitalWrite( PIN_OPEN_OK, LOW);
        outString(menuString[curentMenu], 0,0, 1);
          Serial.print("Found ID #"); Serial.print(finger.fingerID);
          Serial.print(" with confidence of "); Serial.println(finger.confidence); }
      else {
//Якщо пальця немає в БД, вмикається червоний світлодіод
        digitalWrite( PIN_OPEN_ERROR, HIGH);
        outString( "Warning!" ,0,0,1);
        outString( "fingerprint is" ,0,1,0);
        outString( "not found" ,0,2,0); delay(5000);
        digitalWrite( PIN_OPEN_ERROR, LOW);
         outString(menuString[curentMenu], 0,0, 1); } } }
void menuControl() {
  // Перевіряємо стан кнопки "UP"
  if( getButtonState(buttonUpPin) == true) { curentMenu++;
      if( curentMenu > MaxMenuItem){ curentMenu = 0 ; } }
// Перевіряємо кнопки "DOWN"
     if( getButtonState(buttonDownPin) == true ) { curentMenu--;
        if( curentMenu < 0){ curentMenu = MaxMenuItem ; } } }
//Підпрограма входа в меню ( натиснута кнопка "OK")
void runMenuItem(){
  // Перевіряємо кнопки "DOWN"
      if( getButtonState(buttonOkPin) == true ) {
        switch (curentMenu) {
          break;
```

```
        case 1: addNewScan(); break;
        case 2: delScan() ; case 3: showInfo(); } } }
void showInfo() {
  finger.getTemplateCount();
  outString(String(finger.templateCount,10) + " fingerprints", 0,0,1);
  outString("are stored", 0,1,0);
  outString("in a database", 0,2,0); }
//Підпрограма додавання нового відбитка в БД
void addNewScan() {
 String curent = "Curent ID=" ;
 outString( "Add new fingerprint", 0,0, 1 );
 outString( "Select ID", 0,1, 0 );
 outString( curent, 0,2, 0 );
 if( curentID < 1 ){ curentID =1 ; };
 if( curentID > MaxID ){ curentID = MaxID ; }
    for( curentID;curentID<=MaxID;curentID++) {
      if( finger.loadModel(curentID) != FINGERPRINT_OK){ break ; } }
 while( getButtonState(buttonOkPin) != true || getButtonState(buttonReturnPin) !=
true) {
// Якщо натиснута кнопка "OK" додаємо відбиток в БД, якщо натиснута
"RETURN"  - виходимо з додавання
   if( getButtonState(buttonReturnPin) == true ){ curentMenu = 0 ; return ; }
   if( getButtonState(buttonOkPin) == true ){  break ; }
// Якщо натиснуто кнопку "UP" - шукаємо найближчий вільний ID більший за
поточний
   if( getButtonState(buttonUpPin) == true ) { curentID++;
     if( curentID > MaxID ){ curentID = 1 ; };
     for( curentID;curentID<=MaxID;curentID++) {
       if( finger.loadModel(curentID) != FINGERPRINT_OK){ break ; } } }
// Якщо натиснуто кнопку "DOWN" - шукаємо найближчий вільний ID менший
за поточний
   if( getButtonState(buttonDownPin) == true ) { curentID--;
     if( curentID < 1 ){ curentID = MaxID ; }
     for( curentID;curentID>=1;curentID--) {
       if( finger.loadModel(curentID) != FINGERPRINT_OK){ break ; } } }
    outString( String(curentID,10)+ "  ", curent.length(),2,0);
    delay(10); };  outString( curent, 0,0,1 ) ;
 outString( String(curentID,10)+ "  ", curent.length(),0,0); int p = -1;
  while (p != FINGERPRINT_OK) {
   if( getButtonState(buttonReturnPin) == true){ curentMenu = 0 ; break ; }
   p = finger.getImage();
   if( p == FINGERPRINT_OK ){ outString("Image taken",0,0,1);
         p = finger.image2Tz(1);
          if( p == FINGERPRINT_OK ){ outString("Image converted",0,1,0);
          outString("Remove finger",0,2,0); delay(2000);  p = 0;
```

```
          while (p != FINGERPRINT_NOFINGER) {
              p = finger.getImage(); } break ;  } } } p = -1;
  outString("Put the same" , 0,0,1);
  outString("finger again",0,1,0);
  while (p != FINGERPRINT_OK) {
   if( getButtonState(buttonReturnPin) == true){ curentMenu = 0 ; break ; }
    p = finger.getImage();
    if( p == FINGERPRINT_OK ){ outString("Image taken",0,0,1);
      p = finger.image2Tz(2);
    if( p == FINGERPRINT_OK ){ outString("Image converted",0,1,0);
       p = finger.createModel();
    if( p == FINGERPRINT_OK ){ outString("Prints matched!",0,2,0);
       p = finger.storeModel(curentID);
     if (p == FINGERPRINT_OK) { outString("Stored!",0,3,0); curentMenu = 0 ;
        delay(1000); } } } } } }
//Підпрограма видалення відбитка з БД
void delScan(){ String curent = "Curent ID=" ;
  outString( "Delete fingerprint", 0,0, 1 ); outString( "Select ID", 0,1, 0 );
  outString( curent, 0,2, 0 );
      for( curentID;curentID<=MaxID;curentID++) {
       if( finger.loadModel(curentID) == FINGERPRINT_OK){ break ; } }
      if( curentID > MaxID ){ curentID = 1 ; };
 while( getButtonState(buttonOkPin) != true || getButtonState(buttonReturnPin) !=
true) {
    if( getButtonState(buttonReturnPin) == true ){ curentMenu = 0 ; break ; }
    if( getButtonState(buttonUpPin) == true ) { curentID++;
      if( curentID > MaxID ){ curentID = 0 ; };
     while(finger.loadModel(curentID) != FINGERPRINT_OK && curentID <
MaxID){ curentID++;  }; }
    if( getButtonState(buttonDownPin) == true ) { curentID--;
      if( curentID < 0 ){ curentID = MaxID ; };
      while(curentID > 1 && finger.loadModel(curentID) != FINGERPRINT_OK) {
        curentID--; } }
     if( curentID > 0 ){ outString( String(curentID,10)+ "  ", curent.length(),2,0); }
      else {
         outString( "All" , curent.length(),2,0);  }
    if( getButtonState(buttonOkPin) == true ) {
       if( curentID == 0) { delAllScans();
       outString( "All template was del", 0,3,0); }
       else { finger.deleteModel(curentID);
      outString( "Template was deleted", 0,3,0); delay(1000);
      outString( "Delete fingerprint", 0,0, 1 ); outString( "Select ID", 0,1, 0 );
      outString( curent, 0,2, 0 ); }; } delay(10);  }; }
// Видалити всі відбитки
void delAllScans() {
```

```
  for(int i=1;i<=127;i++) {
   if( finger.loadModel(i) == FINGERPRINT_OK){ finger.deleteModel(i); } }
  curentID = 1; }
// Підпрограма отримання стану кнопки
bool getButtonState(int buttonPin){ bool state = false;
    if( digitalRead(buttonPin) == LOW ) {
        // захист від тремтіння контактів
delay(100);
         if( digitalRead(buttonPin) == LOW ){ state = true ; } return state; }
// Підпрограма виводу рядка на екран
void outString(String str, int x, int y, int clear) { if( clear != 0 ){ lcd.clear(); };
  lcd.setCursor(x,y); lcd.print(str); }
```