

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»**

**Радіотехнічний факультет**

**Кафедра радіоприймання та оброблення сигналів**

«На правах рукопису»  
УДК 621.391.8, 621.317.3

До захисту допущено:

В.О. завідувача кафедри

  
А. В. МОВЧАНЮК

« 16 » грудня 2020 р.

**Магістерська дисертація**

**на здобуття ступеня магістра**

**за освітньо-професійною програмою «Радіозв'язок і оброблення сигналів»**

**за спеціальністю 172 «Телекомунікації та радіотехніка»**

**на тему: «3D сканер»**

Виконав:

студент 2 курсу, групи РА-91мп

Шемшур Сергій Юрійович

  
\_\_\_\_\_

Керівник:

ст. викладач. Бондаренко Геннадій Іванович

  
\_\_\_\_\_

Рецензент:

К.т.н., доцент каф. РТПС Шпилька Олександр Олександрович

  
\_\_\_\_\_

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент   
\_\_\_\_\_

Київ – 2020 року

**Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»**

**Радіотехнічний факультет**

**Кафедра радіоприймання та оброблення сигналів**

Рівень вищої освіти – другий (магістерський)

Спеціальність – 172 «Телекомунікації та радіотехніка»

Освітньо-професійна програма «Радіозв'язок і оброблення сигналів»

ЗАТВЕРДЖУЮ

В.О. завідувача кафедри

\_\_\_\_\_ А. В. МОВЧАНЮК

«16» грудня 2020 р.

**ЗАВДАННЯ**

**на магістерську дисертацію студента  
Шемшура Сергія Юрійовича**

1. Тема дисертації «3D сканер»  
науковий керівник дисертації Бондаренко Геннадій Іванович, старший викладач  
затверджені наказом по університету від «5» листопада 2020 р. №3223-с
2. Термін подання студентом дисертації 11 грудня 2020 року
3. Об'єкт дослідження: малогабаритні 3D сканери, принципи побудови та управління.
4. Вихідні дані: керування через Bluetooth, обертання на 360°, можливість надрукувати основні частини на 3D принтері, мати розміри платформи як у 3D сканерів доступних на ринку, живлення від 12 В постійного струму.
5. Перелік завдань, які потрібно розробити: проаналізувати ринок 3D сканерів, визначити можливість створення сканера з мінімізацією витрат на його виготовлення, розробити конструкторську документацію та програму управління 3D сканера, розробити стартап проект впровадження розробки.
6. Орієнтовний перелік графічного (ілюстративного) матеріалу схеми електричні принципи, складальні креслення, креслення деталей.

7. Орієнтовний перелік публікацій \_\_\_\_\_

9. Дата видачі завдання 02 вересня 2020 року

Календарний план

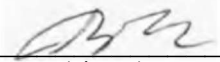
№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Початок	02.09.2020	
2	Розробка вихідних даних	17.09.2020	
3	Огляд ринку сканерів	01.10.2020	
4	Розробка вимог до сканера	15.10.2020	
5	Розробка схем електричних	02.11.2020	
6	Розробка конструкції елементів сканера	16.11.2020	
7	Розробка програмного забезпечення	01.12.2020	
8	Оформлення матеріалів дисертації	10.12.2020	
9	Передача матеріалів для аналізу на плагіат	11.12.2020	

Студент

  
(підпис)

Шемшур С. Ю.  
(ініціали, прізвище)

Науковий керівник дисертації

  
(підпис)

Бондаренко Г. І.  
(ініціали, прізвище)

## РЕФЕРАТ

Магістерська дисертація на тему «3D сканер» складається з 82 сторінок, містить 36 рисунків, 24 таблиці, 6 креслень. До складу пояснювальної записки входить 4 додатки: лістинг програмних кодів, перелік елементів, специфікація друкованого вузла, складальне креслення пристрою.

Метою магістерської дисертації є розробка пристрою для отримання тривимірних моделей реальних об'єктів.

При вирішенні поставленої задачі були проаналізовані технології тривимірного сканування та конкретні сканери. Були визначені їх недоліки, та знайдено вирішення проблеми високої ціни. Головним шляхом вирішення цього є використання методу фотограмметрії та камери смартфона в ролі отримувача візуальної інформації про об'єкт що сканується.

В процесі проектування було використано комплексну систему автоматизованого проектування радіоелектронних засобів Altium Designer, систему автоматизованого проектування SolidWorks. Програмне забезпечення було зроблено за допомогою інтегрованого середовища розробки Android Studio.

Ключові слова: 3D сканування, тривимірна модель, фотограмметрія.

## ABSTRACT

Master's thesis on the topic "3D scanner" consist of 82 pages, contains 36 pictures, 24 tables, 6 drawings and 4 appendices.

The aim of the master's thesis is development of a device for obtaining three-dimensional models of real objects.

In solving this problem were analyzed three-dimensional scanning technologies and specific scanners. Their disadvantages were identified, and a solution to the high price problem was found. The main way to solve this is to use photogrammetry method and a smartphone camera for images capture.

In the design process was used Altium Designer is a PCB and electronic design automation software package for printed circuit boards and SolidWorks a modeling computer-aided design software. The Android device software was developed using the Android Studio integrated development environment.

Keywords: 3D scanning, three-dimensional model, photogrammetry.

## ЗМІСТ

<b>ЗМІСТ</b> .....	4
<b>ПЕРЕЛІК СКОРОЧЕНЬ</b> .....	6
<b>ВСТУП</b> .....	7
<b>1 ОГЛЯД РІШЕНЬ РЕАЛІЗАЦІЇ</b> .....	8
1.1 Огляд існуючих технологій тривимірного сканування .....	8
1.1.1 Контактний метод .....	8
1.1.2 Безконтактний метод .....	8
1.2 Аналіз ринку готової продукції .....	13
1.2.1 Matter and Form V2 .....	13
1.2.2 EINSCAN-SE .....	14
1.2.3 SLS Pro S3 .....	15
1.2.4 Murobo Atlas .....	15
1.3 Огляд можливих удосконалень .....	16
1.3.1 Конструкція .....	16
1.3.2 Управління сканером .....	16
1.3.4 Інтерфейс з 3D принтерами .....	17
1.3.5 Програмне забезпечення .....	17
<b>2 ВИЗНАЧЕННЯ НАПРЯМКУ ДОСЛІДЖЕНЬ ТА РОЗРОБКИ</b> .....	19
2.1 Структурна схема 3D сканера .....	19
2.2 Огляд схеми електричної принципової пристрою управління поворотним столиком .....	20
<b>3 РОЗРОБКА ПОВОРОТНОГО СТОЛИКА</b> .....	21
3.1 Вибір елементної бази .....	21
3.1.1 Вибір мікроконтролера .....	21
3.1.2 Вибір двигуна та драйвера .....	24
3.1.4 Вибір індикатора .....	26
3.1.5 Вибір джерела живлення .....	26
3.1.7 Вибір перемикача .....	28
3.1.8 Вибір з'єднань .....	29
3.1.9 Вибір пасивних компонентів .....	29

3.2 Розрахунок параметрів друкованого монтажу.....	30
3.2.1 Розрахунок площі друкованої плати.....	30
3.2.2 Розрахунок ширин друкованих провідників.....	31
4 КОНСТРУКЦІЯ ПРИСТРОЮ .....	34
5 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ .....	38
5.1 Розробка програмного забезпечення для поворотного столику .....	38
5.2 Розробка програмного забезпечення смартфона на базі Android.....	41
5.3 Програмне забезпечення для реконструкції тривимірних зображень....	42
5.3.1 Meshroom .....	43
5.3.2 Rescap pro .....	46
6 РОЗРОБЛЕННЯ СТАРТАП ПРОЕКТУ .....	47
6.1 Опис ідеї проекту .....	47
6.2 Технологічний аудит ідеї проекту.....	48
6.3 Аналіз ризикових можливостей .....	49
6.4 Розроблення ринкової стратегії проекту .....	53
6.5 Розроблення маркетингової програми стартап-проекту .....	54
ВИСНОВОК.....	56
ВИСНОВКИ.....	57
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	58
ДОДАТОК А.....	61
ДОДАТОК Б .....	74
ДОДАТОК В.....	77
ДОДАТОК Г .....	80

## ПЕРЕЛІК СКОРОЧЕНЬ

ДП – друкована плата.

ПЗ – програмне забезпечення.

BLE – Bluetooth Low Energy.

GATT - General Attribute Protocol

## ВСТУП

З кожним роком технології 3D друку стають більш доступними для звичайних споживачів. І іноді з'являється потреба перенести деякий фізичний об'єкт у віртуальний простір для редагування та подальшого друку на 3D принтері. Якщо об'єкт простий то можна це зробити в ручну, але зі збільшенням складності геометрії це стає важко зробити, тому в таких ситуаціях користуються 3D сканерами.

3D сканування це процес аналізу реальних об'єктів або навколишнього середовища для збирання даних про форму та зовнішній вигляд. Зібрані дані можуть бути використані для створення тривимірних моделей.

3D сканер це периферійний пристрій що дозволяє аналізувати форму об'єкту та на основі отриманих даних конструювати 3D моделі. 3D сканери поділяються на дві категорії: контактні та безконтактні.

Контактний метод оснований на безпосередньому контакті сканера з досліджуваним об'єктом. Безконтактний метод полягає в отриманні даних від об'єкту на відстані. Цей метод є одним способів неруйнівного контролю. Безконтактний сканери бувають двох видів:

- Активні – випромінюють на об'єкт деякі направлені промені і реєструють віддзеркалення для аналізу. Частіше за все використовують лазерний промінь, рідше рентгенівські, інфрачервоні промені або ультразвук.
- Пасивні – не випромінюють, а отримують дані з віддзеркаленого навколишнього випромінення. Більшість сканерів такого типу використовують видиме світло.

## 1 ОГЛЯД РІШЕНЬ РЕАЛІЗАЦІЇ

### 1.1 Огляд існуючих технологій тривимірного сканування

#### 1.1.1 Контактний метод

Контактні 3D сканери досліджують об'єкт через фізичний контакт, наприклад, вимірювання координат точок через сенсори, закріплені на роботизованих «руках» [1]. Такий тип сканерів дуже точний, але займає багато місця та повільний. Також є не нульова можливість пошкодити об'єкт що сканується.



Рисунок 1.1 – контактний 3D сканер

#### 1.1.2 Безконтактний метод

Більшість 3D сканерів на ринку потрапляють саме під цю категорію. Основною перевагою є те, що такі сканери отримують інформацію про об'єкт безконтактно. Поділяються безконтактні сканери на два види: активні та пасивні.

Розглянемо види безконтактних активних сканерів. До таких сканерів належать: сканери на технології вимірювання часу польоту світлового променя, триангуляційні сканери, сканери з структурованим світлом та модульованим світлом:

а) 3D сканери на технології вимірювання часу польоту променя.

В сканерах що вимірюють час польоту променя використовують лазерний промінь для дослідження об'єкта. В основі цієї технології лежить лазерний далекомір. Лазерний далекомір знаходить дистанцію до поверхні вимірюючи час, за який світловий промінь проходить цю відстань. Оскільки швидкість світла відома величина то, якщо знайти час польоту імпульсу світла, можливо знайти відстань. Точність таких сканерів залежить від точності виміру часу польоту, наприклад, для точності в 1 мм необхідно вимірювати час в 3.3 піко секунд.

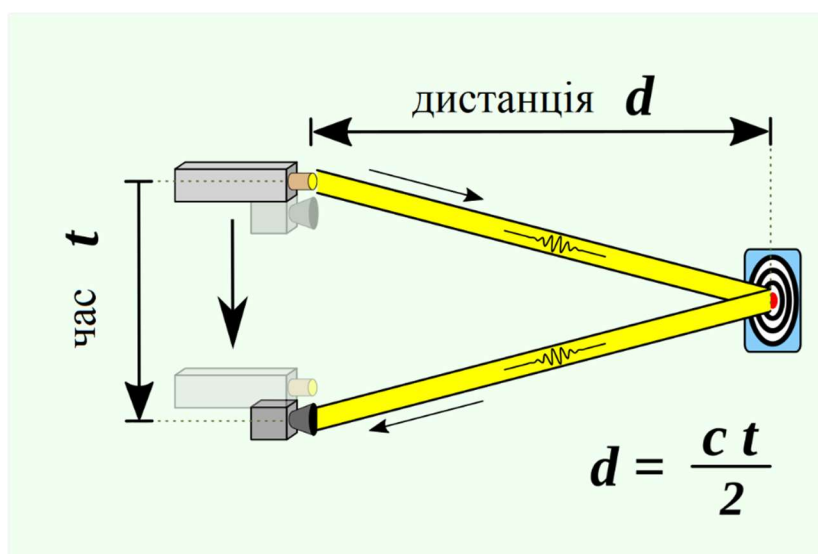


Рисунок 1.2 – Принцип роботи лазерного далекоміра

Лазерний далекомір вимірює відстань лише до однієї точки у просторі, тому для сканування середовища необхідно змінювати напрямок сканування. Це можна досягнути через поворот самого далекоміра, або через набір дзеркал. Сучасні сканери такого типу можуть вимірювати від 10 до 100 тисяч точок в секунду.

Головною перевагою цього типу сканера є можливість роботи на дуже великі відстані, до одиниць кілометрів. Тому підходять більше для сканування великих об'єктів, будівель або ландшафту. Недоліком цього методу є висока ціна компонентів та не висока якість отриманого тривимірного зображення [2].

б) 3D сканери на технології тріангуляції.

Іншим видом активних сканерів є лазерна тріангуляція. Для дослідження об'єкту використовується лазерний промінь. Але на відміну від попередньо розглянутих сканерів для отримання даних використовується камера, яка отримує дані про положення точки на об'єкті. Така техніка називається тріангуляцією через те що лазер, точка на об'єкті та камера формують трикутник.

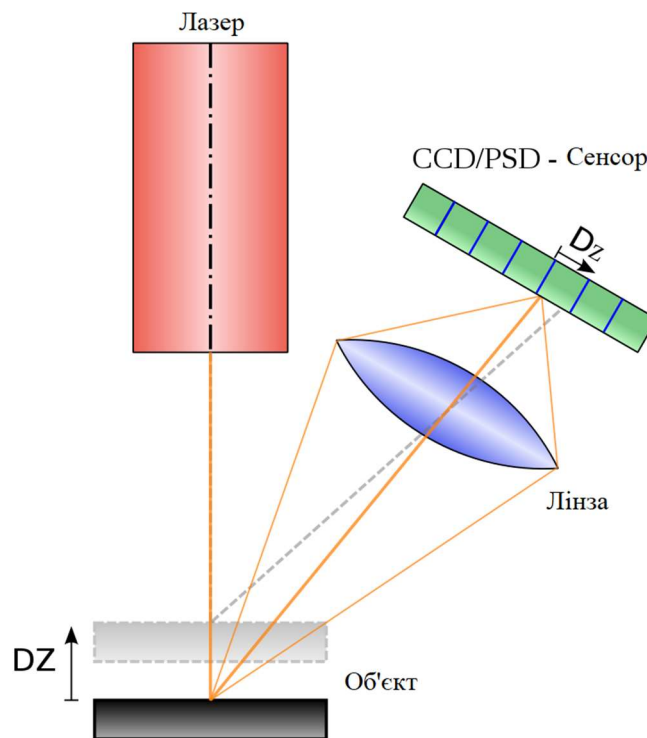


Рисунок 1.3 – Принцип роботи лазерної тріангуляції

Відстань від камери до лазера відома, також відомий кут лазера. Кут камери можна знайти з положення точки на сенсорі. Ці три величини повністю визначають розмір трикутника, та дозволяють знайти позицію лазерної точки на об'єкті [3].

Параметри таких сканерів протилежні до сканерів на технології часу польоту променю, відстані сканування вимірюються до метрів, та можуть дати дуже високу чіткість зображення до десятків мікрометрів.

в) 3D сканери на технології структурованого світла.

Сканери з технології структурованого світла використовують проектор та камеру. Проектор проектує деякий світловий візерунок на об'єкт, а камера,

яка знаходиться на відстані від сканера, отримує викривлене зображення візерунка. На основі отриманих даних комп'ютерна програма будує тривимірне зображення.

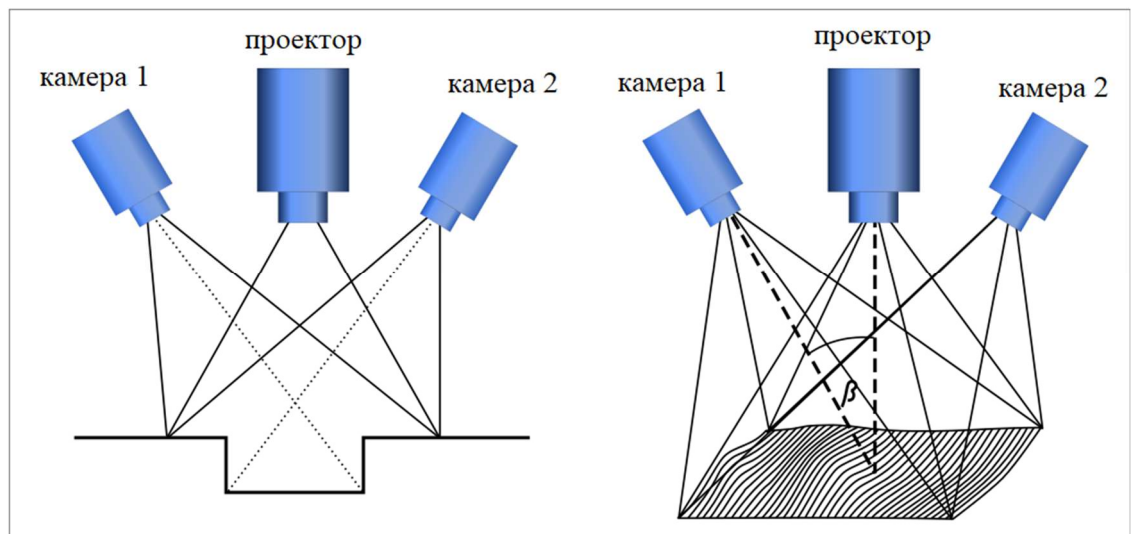


Рисунок 1.4 – Принцип роботи сканерів структурованого світла з двома камерами

Зазвичай використовують візерунок у вигляді паралельних смужок. Типова збірка такого 3D сканера скаладається з одного проєктора та, як мінімум, однієї камери. Для отримання кращих результатів зазвичай використовують дві камери.

У полосах, що спостерігаються, є декілька величин з яких можна визначити глибину зображення. Зміщення будь-якої окремої полоси можна перетворити в тривимірні координати [4].

Розглянемо види безконтактних пасивних сканерів. До таких сканерів належать стереоскопічні, фотометричні та силуетні сканери:

а) Стереоскопічні, фотометричні та силуетні 3D сканери

Ці три види сканерів об'єднує те, що вони нічого не випромінюють на модель, а користуються лише пасивним навколишнім випроміненням. Основною перевагою пасивних методів є те, що в більшості випадків для них треба лише звичайна цифрова камера.

Стереоскопічні системи використовують дві камери на відстані одна від одної. Аналізуючи різницю між двома сигналами можна отримати дані про

відстань до кожної точки об'єкта. Цей метод базується на принципах, що використовується людиною.

Фотометричні системи використовують одну камеру, яка робить знімки об'єкта при різних позиціях освітлення.

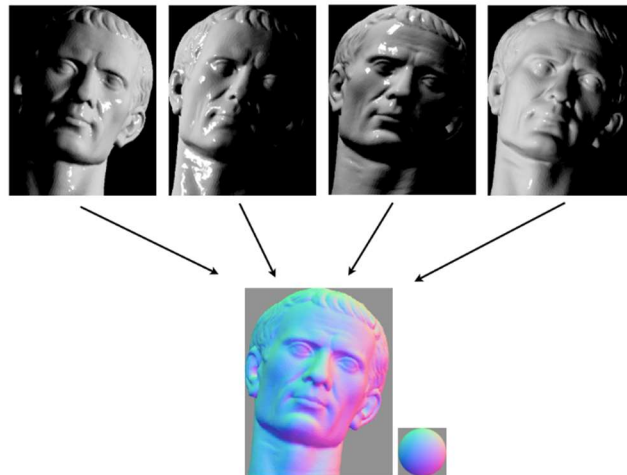


Рисунок 1.5 – Зображення отримане методом фотометрії

Силуетний метод використовує контур створений з набору зображень, зроблених навколо об'єкту на фоні контрастного фону. Ці силуети витягуються, а на їх перетинах створюється апроксимоване зображення об'єкту. Цей метод не дозволяє отримати достовірне зображення об'єктів з впадинами.

#### б) Метод фотограмметрії

Фотограмметрія надає достовірну інформацію про тривимірне положення точок об'єкту, базується на аналізі набору зображень. Тривимірні дані подаються в виді тривимірної хмари точок або тривимірної сітки [5].

Сучасне програмне забезпечення для фотограмметрії дозволяють в автоматичному режимі аналізувати велику кількість цифрових зображень для реконструкції тривимірного зображення. Але в деяких випадках необхідне втручання людини, коли ПЗ не може правильно розставити позиції камер.

Зображення об'єкту робляться з декількох перспектив, задача ПЗ є в тому, щоб по візуальних зображеннях розставити віртуальні камери на місця, де воно було зроблено. Серед такого програмного забезпечення є Autodesk ReCap [6], 3DF Zephyr [7] та інші, платні та безкоштовні.

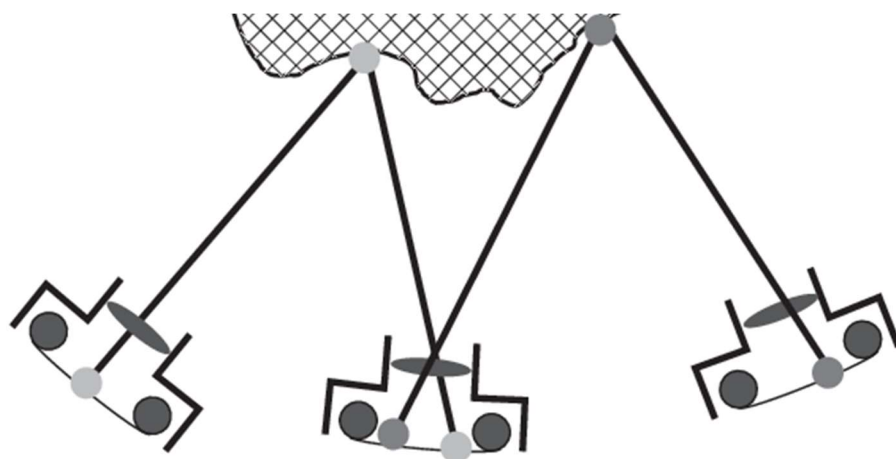


Рисунок 1.6 – Принци роботи фотограмметрії

### Висновки

Розглянувши доступні технології 3D сканування було прийнято рішення про використання методу фотограмметрії. Через його простоту, та доступність ПЗ для реалізації.

### 1.2 Аналіз ринку готової продукції

3D сканери бувають двох видів: ручні та стаціонарні. В даній роботі буде розроблено стаціонарний 3D сканер, тому розглянемо саме такі аналоги.

#### 1.2.1 Matter and Form V2



Рисунок 1.7 – 3D сканер Matter and Form V2

Настільний сканер від компанії Matter and Form розроблений на технології лазерної триангуляції. Це варіант для художників та любителів, які хочуть 3D сканер із високою роздільною здатністю за низьку ціну.

Має такі характеристики:

- Розміри: 345 x 210 x 345 мм;
- Максимальна область сканування: 180 x 180 x 250 мм;
- Вага: 1.7 кг.;
- Роздільна здатність: 0.1 мм;
- Ціна: від \$750.

### 1.2.2 EINSCAN-SE



Рисунок 1.8 – 3D сканер EINSCAN-SE

3D сканер від компанії Shining 3D. Має два режими роботи: автоматичний та ручний. Дозволяє отримувати не тільки форму об'єкта, а і колір поверхні. Безпечний, бо замість лазера використовується звичайне біле світло.

Має такі характеристики:

- Розміри: 570 x 210 x 210мм;
- Мінімальна область сканування: 30 x 30 x 30мм;
- Максимальна область сканування: 700 x 700 x 700 мм;
- Вага: 2.5 кг.;
- Роздільна здатність:  $\leq 0,1$  мм;
- Ціна: від \$1399.

### 1.2.3 SLS Pro S3



Рисунок 1.9 – 3D сканер SLS Pro S3

Професійний 3D сканер від компанії HP. Працює на технології структурованого світла. Технологія базується на проектуванні різних зображень – шаблонів, за допомогою звичайного проектора, а встановлена біля нього камера фіксує отримане викривлення об'єкта, що сканується. Таким чином отримується інформація про форму, глибину і деталі об'єкту, яка потім обраховується, для отримання тривимірної моделі.

Має такі характеристики:

- Максимальна область сканування: 60 – 500 мм;
- Вага: 8.8 кг.;
- Роздільна здатність: до 0.05 мм;
- Ціна: від \$3995.

Хоча даний сканер може давати дуже деталізовані 3D моделі, якщо використовувати їх лише для 3D друку, то маленькі деталі втрачаються. Тому основним недоліком є висока ціна.

### 1.2.4 Murobo Atlas

Доступний настільний 3D сканер що поставляється в виді набору для збирання. Контролюється за допомогою мікрокомп'ютера Raspberry Pi. ATLAS 3D працює, підсвічуючи об'єкт лазером, а потім використовуючи

триангуляцію для створення хмари точок для кожного місця, де лазер потрапляє на модель. Потім сусідні точки з'єднуються утворюючи тривимірну модель.

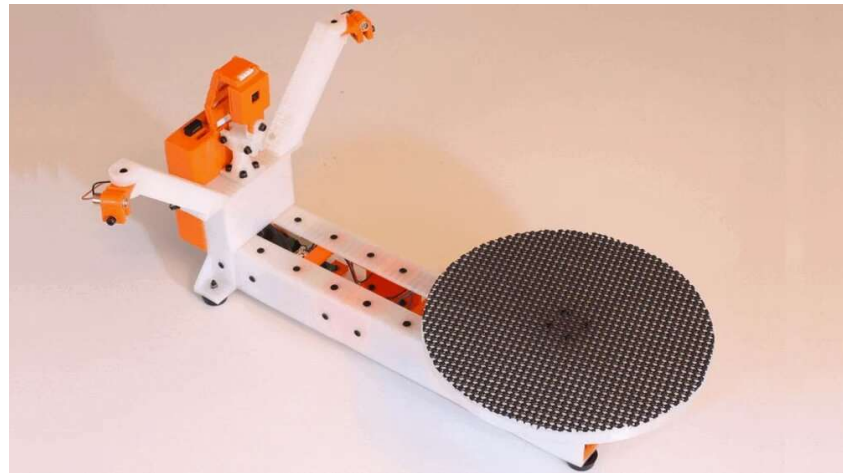


Рисунок 1.10 – 3D сканер Murobo Atlas

Murobo Atlas має такі характеристики:

- Максимальна область сканування: 150 x 250 мм;
- Роздільна здатність: 0.25 мм;
- Час сканування:  $\geq 12$  хв(залежить від вибраної чіткості);
- Роздільна здатність сенсора: 5 мега пікселів;
- Ціна: \$239.

Недоліком цього сканера є сенсор не великої роздільної здатності що використовується для отримання зображень об'єкту. Також недоліком є те що сканер поставляється без корпусу, який користувач повинен буде зробити сам на 3D принтері, або докупити окремо.

### 1.3 Огляд можливих удосконалень

#### 1.3.1 Конструкція

Механічна частина 3D сканера повинна бути такою, яку легко було б надрукувати на 3D принтері. Буде використаний схожий на аналоги поворотний стіл, що керується кроковим двигуном.

#### 1.3.2 Управління сканером

Все керування сканером можна буде виконувати через додаток на смартфоні. Тобто запуск сканування, його відміна та налаштування. Під'єднання

буде реалізовано через протокол Bluetooth low energy. Він має низьке енергоспоживання, через що сканер можна буде живити за допомогою вбудованої батареї, а не тільки через мережений блок живлення.

Також на обертовому столику буде встановлено дисплей, який буде виводити данні для підключення смартфона по Bluetooth, а також сервісну інформацію про стан сканера.

### 1.3.3 Камера для сканування

В сканері буде використовуватися технологія фотограмметрії, тобто реконструкція тривимірної моделі за набором двовимірних зображень. Від якості зображень з камери залежить якість отриманих тривимірних моделей. В якості камери буде виступати камера смартфона, які в останній час комплектуються сенсорами з високою роздільною здатністю.

### 1.3.4 Інтерфейс з 3D принтерами

Необхідно буде розробити таку систему, що видавала б відсканований об'єкт у такому форматі, що дозволяв би без модифікацій відправляти модель на друк 3D принтером.

### 1.3.5 Програмне забезпечення

Необхідно буде розробити програмне забезпечення для мікроконтролера встановленого в поворотному столику. Також необхідно розробити додаток для смартфона на операційній системі Android. Він буде керувати платформою та робити зображення. Операційна система Android обрана через більшу доступність таких смартфонів та через набагато простішу розробку додатків під неї.

Необхідно буде вибрати програмне забезпечення для реконструкції тривимірних моделей методом фотограмметрії.

## ВИСНОВКИ ДО ПЕРШОГО РОЗДІЛУ

Розглянувши 3D сканери доступні на ринку можна бачити що вони доступні в доволі широкому ціновому діапазоні. Всі сканери є активними, тобто випромінюють промені на об'єкт і за відображенням реконструюють 3D модель. Більшість мають обертовий столик, що дозволяє отримати модель за одне сканування без перестановки моделі. Час сканування варіюється від 12 хвилин (Murobo Atlas) до 1 години (SLS Pro S3) та залежить від вибраної роздільної здатності сканування, яка лежить в межах від 0.25 мм до 0.05 мм. Всі сканери поставляються зі своїм програмним забезпеченням.

Основним недоліком розглянутих сканерів є їх висока ціна. Тому основною метою роботи буде розробка доступного 3D сканера, що дозволить розширити аудиторію користувачів.

## 2 ВИЗНАЧЕННЯ НАПРЯМКУ ДОСЛІДЖЕНЬ ТА РОЗРОБКИ

Після огляду аналогічних пристроїв було прийнято рішення використувати схожу конструкцію, тобто майбутній пристрій буде складатися з деякого сенсору візуальної інформації та поворотного столику на якому буде знаходитися об'єкт, що сканується.

### 2.1 Структурна схема 3D сканера

Розглянемо спочатку структурну схему майбутнього пристрою.

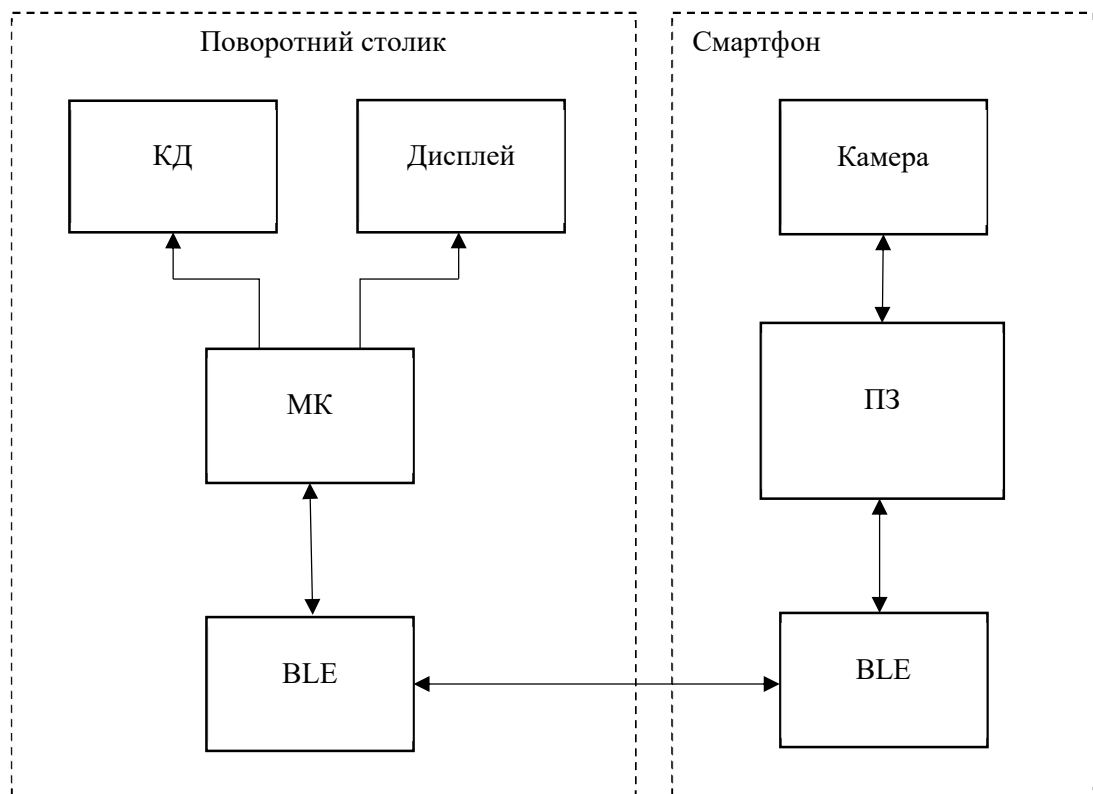


Рисунок 2.1 – Структурна схема 3D сканера

МК – мікроконтролер з вбудованим Bluetooth.

КД – кроковий двигун, що обертає платформу.

Дисплей – рідкокристалічний дисплей для виводу інформації.

BLE – модуль Bluetooth low energy.

Камера – камера смартфона для отримання інформації про об'єкт.

ПЗ – програмне забезпечення, що керує камерою та поворотним столиком через BLE

## 2.2 Огляд схеми електричної принципової пристрою управління поворотним столиком

Для роботи поворотного столику необхідно буде розробити пристрій управління, який буде отримувати сигнали керування та повертати об'єкт, що сканується. Розглянемо його схему електричну принципову.

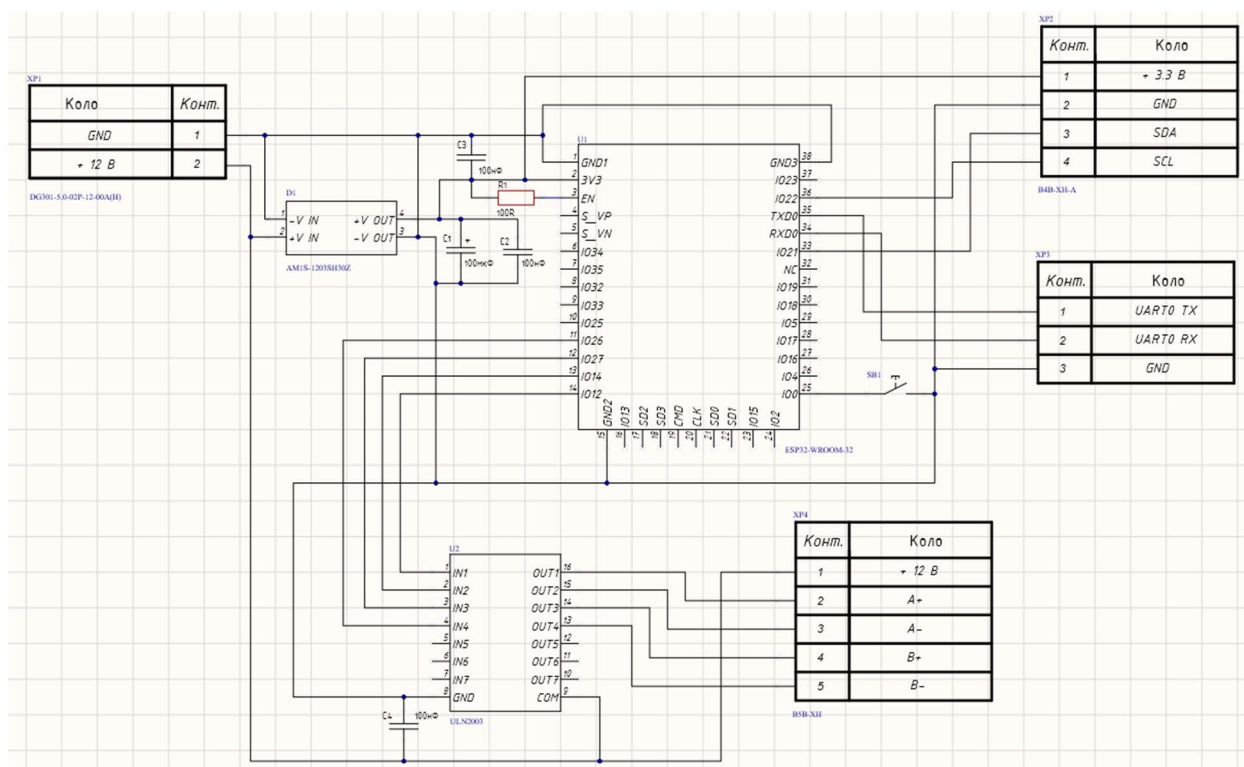


Рисунок 2.2 – Схема електрична принципова пристрою управління поворотним столиком

Можна бачити зі схеми, що головною частиною пристрою управління є мікроконтролер. Він отримує команди по бездротовому протоколу Bluetooth, обробляє їх та керує кроковим двигуном та індикатором.

Для підведення живлення використовується клемне з'єднання. Для під'єднання індикатора та крокового двигуна передбачені штекери. Для програмування виведені відповідні сигнали на штекер. Для переведення в режим програмування використовується кнопка.

На виході понижуючого перетворювача та біля ніжок живлення мікросхем встановлені фільтруючі конденсатори.

## 3 РОЗРОБКА ПОВОРОТНОГО СТОЛИКА

Однією із складових частин 3D сканера є поворотний столик. Від дозволяє сканувати весь об'єкт без необхідності його переставляти вручну.

Після огляду аналогів можна сформулювати такі вимоги:

- керування через Bluetooth
- живлення від 12 В постійного струму
- можливість надрукувати основні частини на 3D принтері
- мати розміри платформи як у 3D сканерів доступних на ринку
- обертання на 360°

Для керування поворотним столиком необхідно буде використовувати мікроконтролер з внутрішнім, або зовнішнім модулем Bluetooth. Це дозволить керувати ним з будь – якого пристрою з Bluetooth.

### 3.1 Вибір елементної бази

#### 3.1.1 Вибір мікроконтролера

Мікроконтролер є основною складовою частиною поворотного столика. Його задачею буде отримання команд через Bluetooth, їх обробка та виконання, а також виведення сервісної інформації на індикатор.

Серед популярних мікроконтролерів є серії AVR від Atmel, STM від STMicroelectronics, PIC від Microchip. Але основним їх недоліком є те, що при їх використанні необхідно також буде додати модуль безпроводного інтерфейсу Bluetooth, що підвищило складність конструкції, та збільшило її ціну.

Саме тому в ролі серця пристрою управління було обрано мікроконтролер ESP32 від компанії Espressif Systems.

Для розробки пристрою управління була обрана найпростіша версія цього мікроконтролера, доступна на даний час, ESP32-WROOM-32.

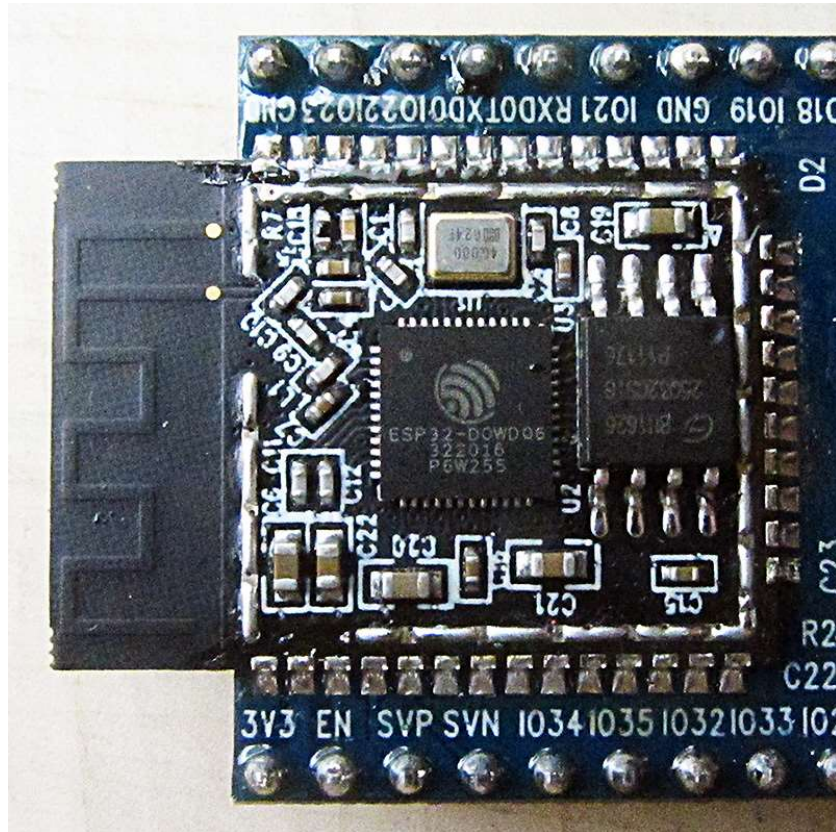


Рисунок 3.1 – Мікроконтролер ESP32-WROOM-32

Мікроконтролер ESP32-WROOM-32 має такі параметри [8]:

- Напруга живлення: 3.0 – 3.6 В
- Середній споживаний струм: 80 мА
- Частота вбудованого кварцового кристалу: 40 МГц
- Bluetooth: v4.2 BR/EDR і BLE
- Wi-Fi: 802.11 b/g/n
- 12 розрядний АЦП до 18 каналів
- Два 8 бітні ЦАП
- 4 × SPI майстер інтерфейси
- 2 × I<sup>2</sup>C майстер інтерфейси
- 3 × UART інтерфейси

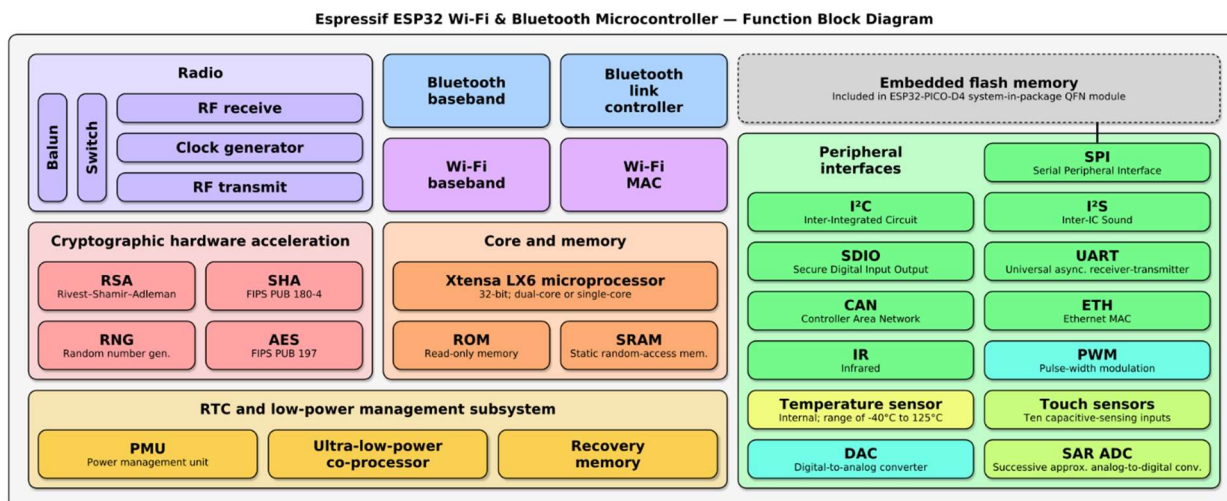


Рисунок 3.2 – Блок схема ESP32

Цей мікроконтролер було обрано через вбудований Bluetooth, високу продуктивність роботи, та відносно низьку ціну відносно інших, схожих мікроконтролерів, при тому до них додатково потрібно було встановлювати Bluetooth модуль.

В ролі антени Bluetooth та Wi-Fi в цьому мікроконтролері використовується планарна антена. Це необхідно буде врахувати при проектуванні друкованої плати, через те, що проходження доріжок на платі біля антени можуть змінити її параметри.

Оскільки мікроконтролер поставляється в виді модуля для поверхневого монтажу, в нього вже вбудовано кварцовий резонатор, пам'ять довільного доступу та обв'язка.

При проектуванні друкованої плати необхідно буде передбачити з'єднання на платі для програматора, а також можливість подачі на 25 ніжку (IO0) мікроконтролера сигналу високого рівня для переведення його в режим прошивки. Також для включення мікроконтролера при подачі напруги необхідно підтягнути 3 ніжку (EN) до високого рівня сигналу.

Також великою перевагою цього мікроконтролера є можливість програмувати його різними мовами програмування C/C++ або Python [9]. Саме через можливість розробки програмного забезпечення мовою Python зменшується час від ідеї до готового пристрою.

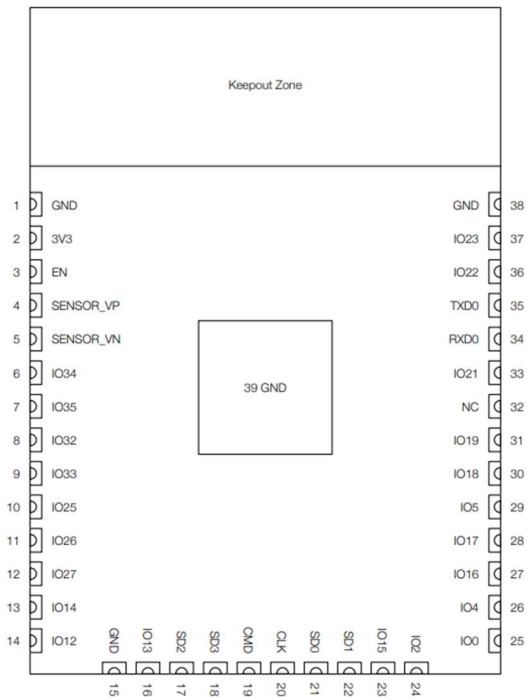


Рисунок 3.3 –Розташування виводів ESP32-WROOM-32

### 3.1.2 Вибір двигуна та драйвера

В ролі двигуна що виконує поворот обертової платформи було обрано кроковий двигун 28BYJ-48 [10].



Рисунок 3.4 –Кроковий двигун 28BYJ-48

Кроковий двигун був обраний через можливість точно керувати кутом повороту, а наявний редуктор дозволяє отримати більший крутячий момент та дозволить сканувати предмети більшої ваги.

Параметри крокового двигуна такі:

- Кількість кроків на повний поворот: 32
- Співвідношення понижуючого редуктора: 1/16
- Напруга живлення: 12 В
- Крутний момент: 34.3 Н/м
- Піковий струм однієї обмотки: 320 мА

Оскільки обраний мікроконтролер з одного виводу може дати лише 12мА [11] необхідно використовувати драйвер який керуватиме кроковим двигуном.

3.1.3 В ролі драйвера було обрано мікросхему ULN2003, а саме її модифікацію ULN2003D в корпусі SOIC для поверхневого монтажу.

Мікросхема ULN2003D це збірка з семи транзисторів Дарлінгтона, розрахованих на відносно високу напругу та струм.

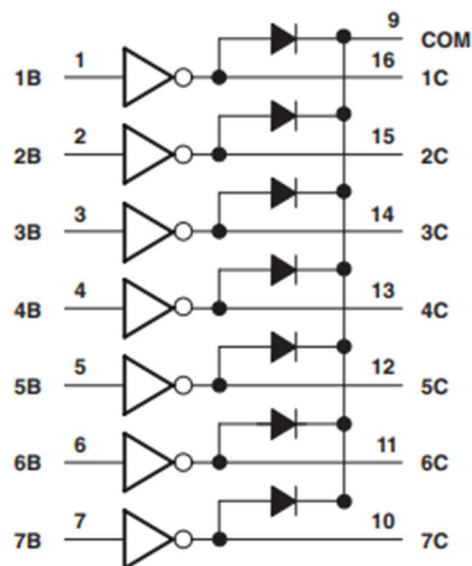


Рисунок 3.5 – Спрощена діаграма мікросхеми ULN2003

Параметри мікросхеми ULN2003 є такими [12]:

- Кількість драйверів в мікросхемі: 7
- Максимальна вихідна напруга: 50 В
- Максимальний вихідний струм: 500 мА (на один канал)
- Сумісність з вхідними рівнями: ТТЛ та КМОП

### 3.1.4 Вибір індикатора

Для індикації роботи, та виводу сервісної інформації було обрано дисплей. Ідеальним для такої задачі буде дисплей на органічних світлодіодах з драйвером SSD1306.



Рисунок 3.6 – OLED дисплей SSD1306

Параметри дисплею такі [13]:

- Роздільна здатність: 128 x 64 точок
- Діагональ екрану: 2.4 см
- Напруга живлення: 3.3 – 6 В
- Інтерфейс керування: I<sup>2</sup>C

### 3.1.5 Вибір джерела живлення

Оскільки кроковий двигун живиться від 12 В постійного струму, для живлення 3D сканера було обрано мережевий блок живлення 12В 3А, 5.5x2.1мм [14].



Рисунок 3.7 – Мережевий блок живлення

Для підключення блоку живлення до 3D сканера необхідно вилку живлення. Було обрано DC-022В під розетку 5.5/2.5мм.



Рисунок 3.8 – Мережевий блок живлення

3.1.6 Оскільки мікроконтролер живиться від 3.0 В до 3.6 В, а індикатор живиться від 3.3 В до 6 В постійного струму було зроблено висновки про необхідність використання понижуючого перетворювача.

Спочатку розрахуємо струм яким живиться мікроконтролер та індикатор.

Максимальний споживаний струм мікроконтролера ESP32-WROOM-32 при передачі по протоколу BLE:  $I_{\text{МК}} = 80 \text{ мА}$

Максимальний споживаний струм індикатора SSD1306 [15]:  $I_{\text{інд}} = 20 \text{ мА}$

Сумарне споживання:  $I_{\Sigma} = I_{\text{МК}} + I_{\text{інд}} = 100 \text{ мА}$

Для живлення низьковольтної частини 3D сканера було обрано dc – dc перетворювач AM1S-1203SH30Z від компанії AIMTEC.



Рисунок 3.9 – Понижуючий перетворювач AM1S-1203SH30Z

Він має такі параметри [16]:

- Вхідна напруга: 10.8 – 13.2 В
- Вихідна напруга: 3.3 В
- Максимальний струм на виході: 303 мА
- Вхідний струм (під навантаженням/без навантаження) 107/15 мА
- Ефективність: 80%

### 3.1.7 Вибір перемикача

Для вмикання та вимикання пристрою на передній панелі було встановлено перемикач КСD-1-104.



Рисунок 3.10 – Перемикач КСD-1-104

Характеристики перемикача КСD-1-104 є такими:

- Номінальна напруга: 250В
- Номінальний струм: 6А
- Опір контактів: не більше 35 мОм
- Фіксація: є
- Робочий ресурс: 10000 циклів

### 3.1.8 Вибір з'єднань

Для приєднання крокового двигуна та індикатора було обрано використовувати роз'ємні з'єднання типу ХН [17], оскільки кроковий двигун поставляється з таким з'єднанням.

Для під'єднання крокового двигуна було обрано вилку В5В-ХН-А, яка має 5 контактів, а для індикатора обрано вилку В4В-ХН-А, що має 4 контакти.



Рисунок 3.11 – Вилка В5В-ХН-А

Для під'єднання до мікроконтролера, його програмування та відлагодження було виведено апаратний UART0 на окреме з'єднання типу РН на 3 контакти.

### 3.1.9 Вибір пасивних компонентів

Для автоматизації виробництва було обрано пасивні компоненти поверхневого монтажу.

В ролі блокуючих конденсаторів було обрано один танталовий конденсатор ТАJD107M010RNJ для згладжування струму на виході понижуючого перетворювача, всі інші конденсатори це керамічні СС1206KRX7R9ВВ104.

Підтягуючий резистор для подання високого рівня на вхід мікроконтролера було обрано RCS1206100KFKEA.

### 3.2 Розрахунок параметрів друкованого монтажу

При проектуванні 3D сканера постала задача розробити друковану плату для пристрою управління поворотним столиком. Розробка друкованої плати буде проходити в системі автоматизованого проектування Altium Designer. Спочатку необхідно розрахувати параметри друкованого монтажу.

Матеріалом основи друкованої плати було обрано FR-4 35/35 1.5. Це композитний матеріал, що складається з шарів склопластику фольгованого з двох сторін міддю.

Методом виготовлення було обрано комбінований негативний метод. Він полягає у використанні фольгованого діелектрика, з якого хімічним методом знімається шар фольги в місці, де немає доріжок. Після цього свердяться та металізуються отвори. Наноситься захисний шар паяльної маски на всю площу плати крім контактних площадок. Після чого контактні площадки покриваються захисним шаром олова.

Було обрано 3-й клас точності. 3-й клас точності ДП для мікросхем з планарними і штирьовими виводами з середнім насиченням поверхні плати елементами.

#### 3.2.1 Розрахунок площі друкованої плати

Для розрахунку площі друкованої плати необхідно врахувати розміри всіх елементів плати.

*Таблиця 3.1 Розміри і площа елементів*

Елемент	Довжина, мм	Ширина, мм	Кількість, шт	Площа, мм <sup>2</sup>	Коеф.
Мікроконтролер ESP32	25.5	18	1	1377	3
Понижуючий перетворювач AM1S-1203SH30Z	11.7	6	1	210	3

Таблиця 3.1 (Продовження)

Транзисторна збірка ULN2003D	9,9	3,9	1	107	3
Клемник DG301	10	7,6	1	114	1,5
Вилка B5B-ХН-А	14,9	5,75	1	129	1,5
Вилка B4B-ХН-А	12,4	5,75	1	106	1,5
Конденсатор TAJD107M010RNJ	7,3	4,3	1	47	1,5
Кнопка IT-1187-160GM-G	6,4	5,2	1	50	1,5
Конденсатор CC1206KRX7R9BB104	3,2	1,6	3	23	1,5
Резистор RCS1206100KFKEA	3,2	1,6	1	7,6	1,5
Отвори кріплення	3	3	4	28	1

Сумарна площа всіх компонентів виходить 2200 мм<sup>2</sup>. Візьмемо плату розмірами 60 на 50 мм, площа якої 3000 мм<sup>2</sup> що більше розрахованого, але доцільніше взяти більшу.

### 3.2.2 Розрахунок ширин друкованих провідників

Розрахуємо ширину сигнальних провідників.

Мінімальна товщина провідників розраховується за формулою:

$$t_{1mi} = t_{min} + |\Delta t_{н.о}| \quad (3.1)$$

Де  $t_{min}$  - мінімальне значення номінальної ширини провідника;

$\Delta t_{н.о}$  – допуск на ширину провідника;

$$t_{min} = 0,25;$$

$$\Delta t_{н.о} = 0,05$$

$$t_{1min} = 0,25 + |0,05| = 0,3 \text{ мм}$$

Визначимо ширину силових доріжок до крокового двигуна. Розрахуємо ширину провідника на друкованій платі виходячи з струму що в ньому протікає.

$$t = \frac{I_{max}}{h_p \cdot j_p} \quad (3.2)$$

Де  $I_{max}$  - максимальний струм в провіднику;

$h_p$  – товщина шару металізації;

$j_p$  – допустима щільність струму в провіднику

$I_{max}$  в колі крокового двигуна при максимальному навантаженні рівний близько 300 мА. Товщина металізації  $h_p$  для матеріалу FR-4 35/35 1,5 рівна 35 мікрометрів.

$$j_p = 20 \text{ А/мм}^2$$

Товщина силових доріжок:

$$t = \frac{0,3}{35 \cdot 10^{-3} \cdot 20} = 0,43 \text{ мм}$$

Отримані після розрахунків ширини доріжок:

Силові: 0,5 мм

Сигнальні: 0.3 мм

### 3.2.3 Проектування друкованої плати

Проектування друкованої плати для пристрою управління було виконано в системі автоматизованого проектування Altium Designer.

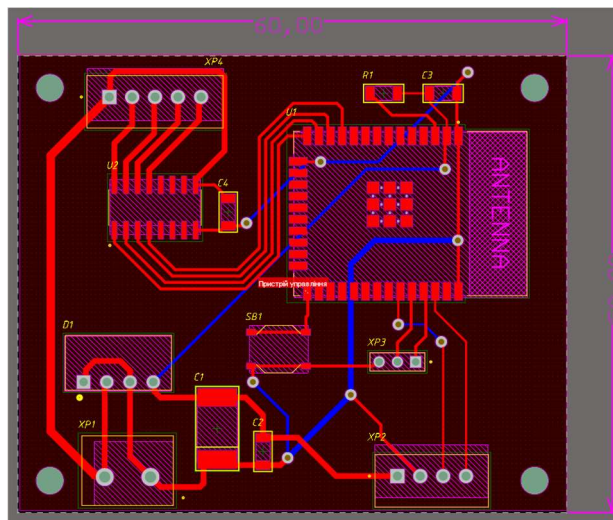


Рисунок 3.12 – Спроектвана ДП

Також за допомогою цього програмного забезпечення було спроектовано схему електричну принципову, креслення друкованої плати та друкованого вузла.

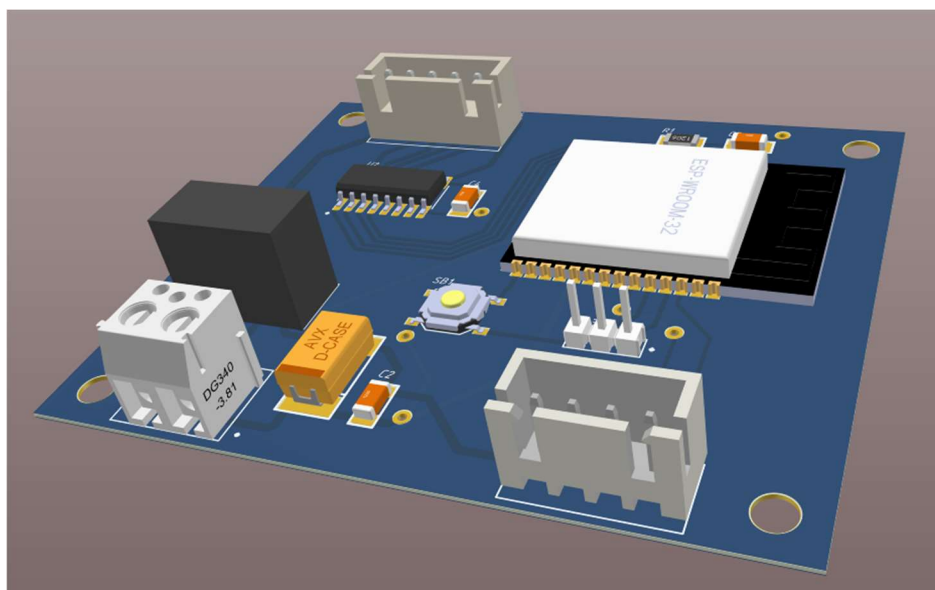


Рисунок 3.13 – Тривимірний модель спроектованої ДП

## 4 КОНСТРУКЦІЯ ПРИСТРОЮ

Для пришвидшення розробки мінімально робочої моделі було зроблено висновок про необхідність використання сервісів по продажу тривимірних моделей в мережі Інтернет. На платформі CGTrader [18] було зроблено покупку ліцензії на використання та модифікацію схожого по функціоналу пристрою, обертового столику для фотографування.



Рисунок 4.1 – Куплений пристрій

Після покупки було зроблено рішення про модифікацію тривимірної моделі під конкретну задачу. Головні зміни це кріплення під розміщення друкованої плати та змінені органи управління.

Було прийнято рішення що поворотний столик та смартфон будуть не закріплені один відносно іншого, що дає користувачу більшу свободу в виборі позиції сканування. Також з широким розповсюдженням 3D принтерів було зроблено рішення друкувати основні деталі корпусу на принтері. Використання адитивного методу виробництва дало доступ до технік де доступних для традиційних технологій.

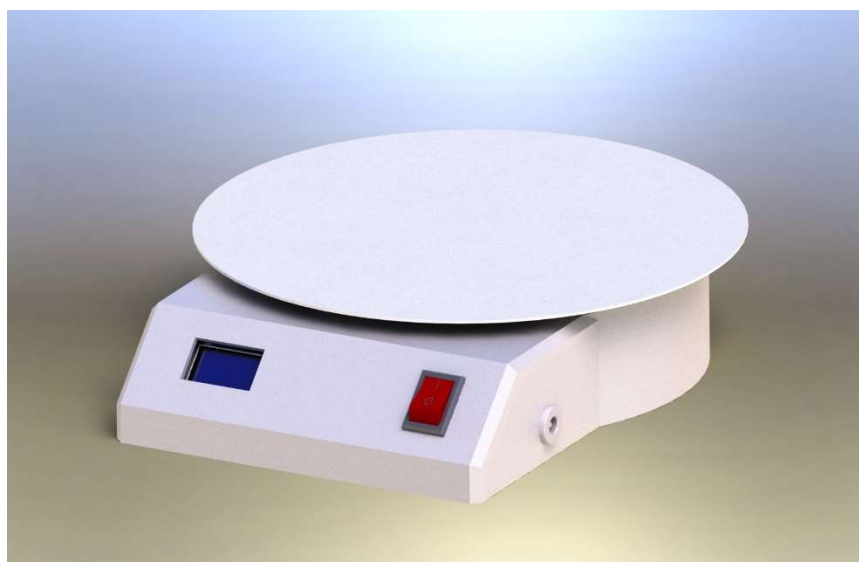


Рисунок 4.2 – Тривимірна модель модифікованого поворотного столику

На рисунку 4.2 можна бачити тривимірну модель модифікованого обертового столику. На передній панелі індикатор для виводу службової інформації про статус 3D сканера, перемикач та штекер живлення.

Матеріалом корпусу було обрано термопластик PLA. Такий матеріал було обрано через відносну простоту друку та фізичні властивості [19].

Таблиця 4.1 Параметри термопластику PLA

Величина	Значення
Температура плавлення	173 – 178 °C
Твердість (по Роквеллу)	R70 – R90
Міцність на згин	55,3 МПа
Міцність на розрив	57,8 МПа
Густина	1,25 г/см <sup>3</sup>
Мінімальна товщина стінок	1 мм
Точність друку	± 0,1 %
Розмір найменших деталей	0,3 мм
Усадка при виготовленні деталей	менше 0,8%

Розглянемо внутрішню будову модифікованого поворотного столику.

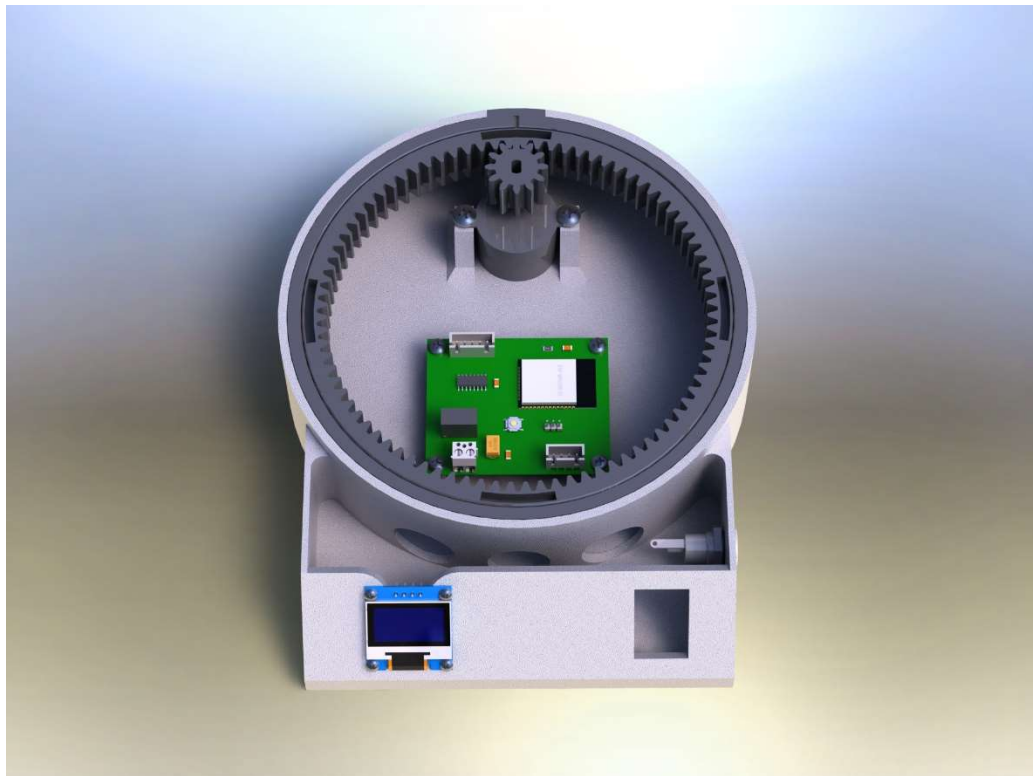


Рисунок 4.3 – Внутрішня будова поворотного столику

Всі органи керування, крім перемикача живлення винесено в додаток для операційної системи Android. Принцип роботи якого буде розглянути в наступному розділі.

Головною частиною поворотного столика є ДП, що була розроблена в попередньому розділі. 12 В живлення з мережевого блоку через штекер та перемикач приходять на плату. Живлення потрапляє на драйвер крокового двигуна та через понижуючий перетворювач на мікроконтролер. Який в свою чергу керує індикатором та кроковим двигуном.

Кроковий двигун, обертає шестерню, яка в свою чергу обертає колесо на якому встановлена платформа з об'єктом що сканується. Використання 3D друку дозволяє виробляти деталі недоступні для інших технологій виробництва.

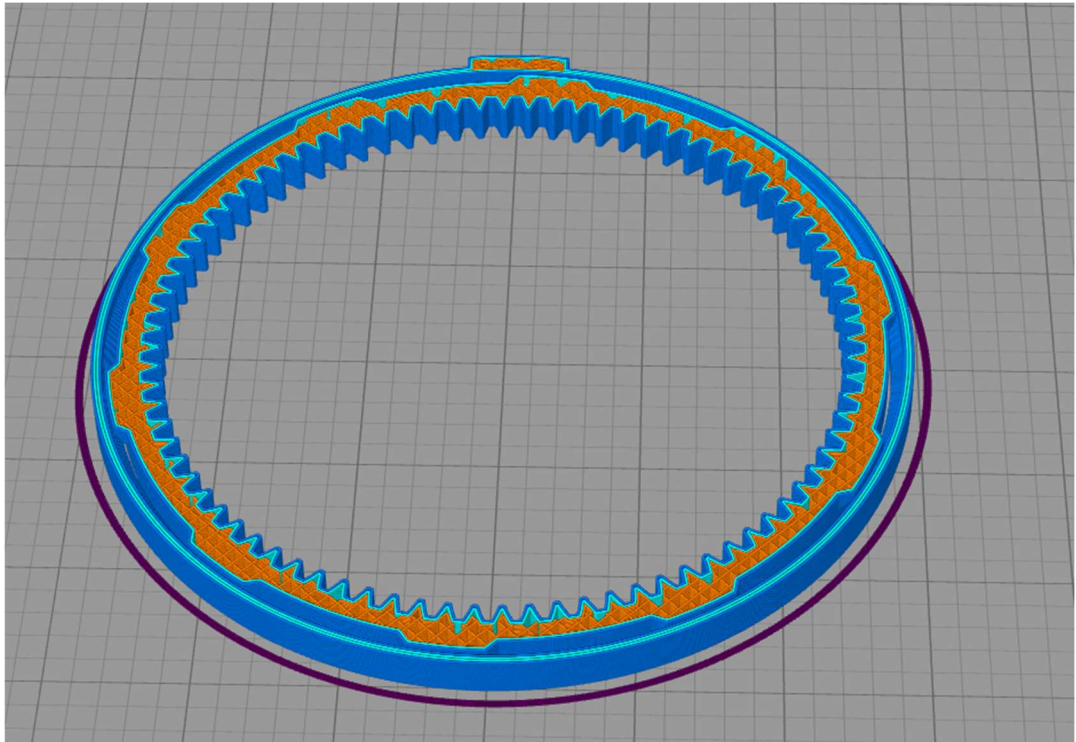


Рисунок 4.3 – Внутрішня будова друкованого підшипника

На рисунку 4.3 зображено приклад деталі яку неможливо точно відтворити без використання адитивного виробництва. Вона представляє собою зовнішнє кільце з пазом, по якому ковзає внутрішнє кільце. Проміжок між ними складає 0,5 мм. Друкуються вони однією деталлю. А оскільки можна регулювати внутрішнє заповнення деталей можна зробити їх легшими, що зменшує витрату пластику не сильно впливаючи на міцність.

## 5 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Після розробки ДП та корпусу постала необхідність розробки ПЗ для керування 3D сканером та реконструкцією тривимірного зображення.

### 5.1 Розробка програмного забезпечення для поворотного столику

Для мікроконтролерів серії ESP ПЗ розробляється в спеціалізованому програмному середовищі `esp-idf` [20]. Але для пришвидшення проектування було використано середовище `MicroPython` для ESP32 [21].

`MicroPython` це швидка та ефективна реалізація мови програмування Python 3. Вона включає в себе невеликий набір бібліотек та оптимізована під роботу в обмежених середовищах, як на мікроконтролерах.

Головною перевагою цієї реалізації є те що немає необхідності компілювати програмний код, він може бути звантажений на мікроконтролер напямую, навіть через безпроводний інтерфейс. Це дозволяє сильно пришвидшити розробку, на прискорити вихід пристрою на ринок.

Програмний код напямую виконується на мікроконтролері, тому є деяка втрата продуктивності. Але через те, що мікроконтролер ESP32-WROOM-32 має відносно велику продуктивність, то на невеликих програмах, не критичних до швидкості виконання, це не є проблемою. Саме тому для програмування мікроконтролера було обрано мову Python 3

На рисунку 5.1 зображено спрощену схему алгоритму роботи програми поворотного столика.

При появі живлення мікроконтролер автоматично починає налаштування виходів. Виконує ініціалізацію індикатора, драйвера крокового двигуна та BLE. Після ініціалізації мікроконтролер починає анонс своїх параметрів, для того щоб зовнішні пристрої його побачили та могли встановити з'єднання [22].

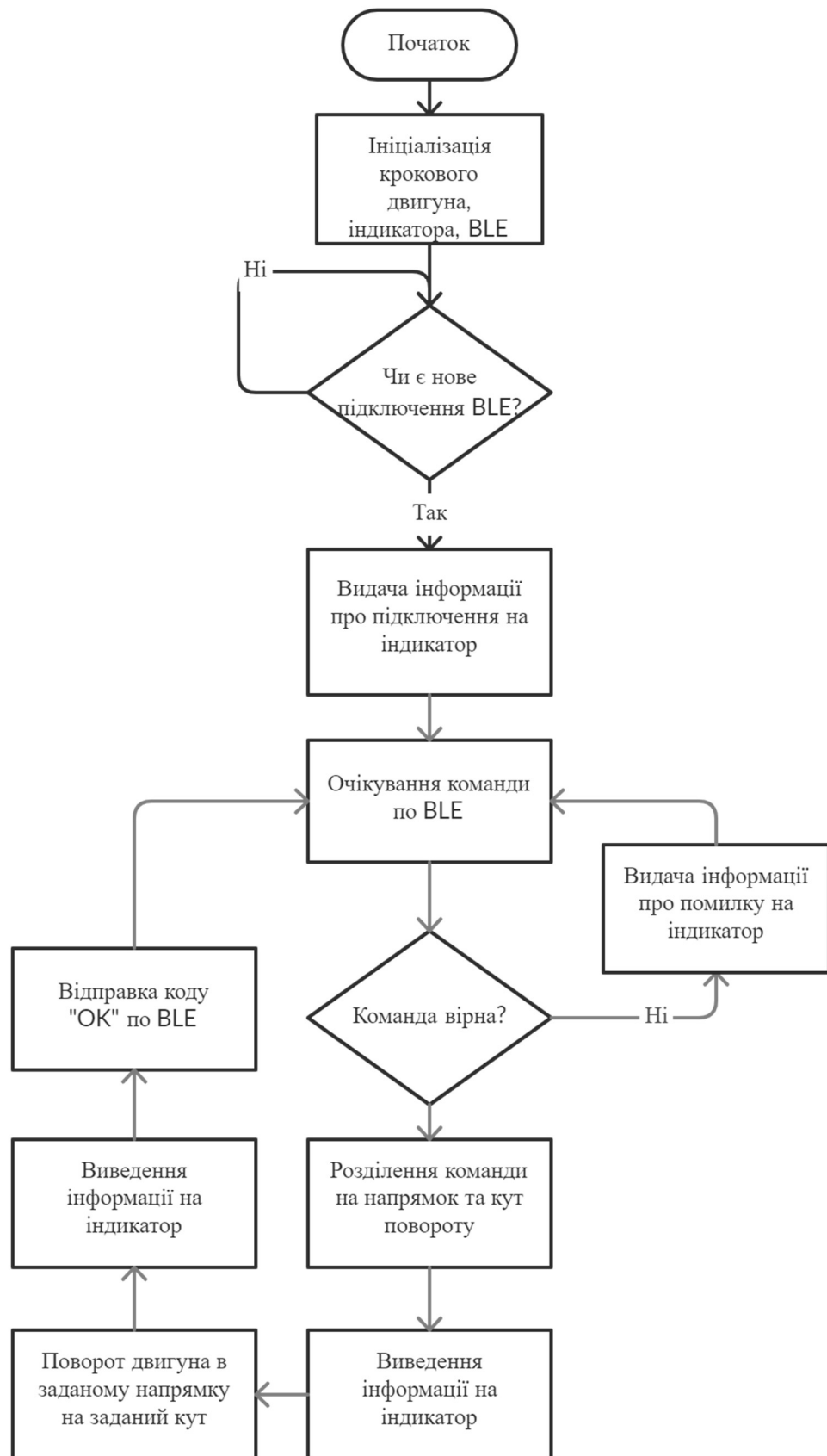


Рисунок 5.1 – Алгоритм роботи поворотного столика

Протокол BLE вимагає щоб периферійний пристрій обов'язково повинен реалізовувати профіль GATT.

У профіля GATT може бути декілька сервісів, що дозволяє в одному пристрої поєднувати декілька різних за типом приладів. Сервіс - це набір характеристик для конкретного призначення. Це необхідно для уніфікації роботи з різними приладами одного виду. Наприклад, носимі пульсометри, всередині вони можуть бути побудовані по різному, але це не важливо, бо вони реалізуються єдиний сервіс, що описує, як повинні поводити себе пристрої BLE

У кожного сервісу та характеристики є свій унікальний 128-бітний ідентифікатор. Поворотний столик реалізує сервіс Nordic UART [23]. Цей сервіс емулює протокол UART і дозволяє обмінюватися довільними даними.

В сервісу Nordic UART наявні дві характеристики:

- RX Characteristic (UUID: 6E400002-B5A3-F393-E0A9-E50E24DCCA9E)
- TX Characteristic (UUID: 6E400003-B5A3-F393-E0A9-E50E24DCCA9E)

RX характеристика використовується для передачі даних до мікроконтролера, а TX відповідно для отримання даних від мікроконтролера. Також протокол дозволяє підписуватися на зміни характеристик і він повідомить, коли якась із них змінилася, тому немає необхідності постійно перевіряти, чи є нові дані.

Після того як мікроконтролер отримає запит на підключення він його приймає та починає очікувати команди. Команда представляє з себе текстовий рядок виду «Tx-n», де x – напрямок повороту (R – по часовій стрілці, L – проти), n – кількість градусів, на які необхідно повернути платформу.

Після отримання рядку команди програма розділяє його на дві частини. На напрямок повороту та величину. Потім обчислюється необхідна кількість кроків і виконується поворот платформи.

Після повороту платформи в RX характеристику записується значення «ОК», що повідомляє про готовність приймати наступні команди. Якщо під час повороту платформи приходить нова команда то вона додається в чергу і виконається по закінченню повороту.

Програмний код до мікроконтролера наведений в додатку А

## 5.2 Розробка програмного забезпечення смартфона на базі Android

Після розробки програмного забезпечення для поворотного столику по-стала задача розробки додатка для операційної системи Android. Вона обрана через доступність засобів розробки, на відміну від системи iOS. Проектування ПЗ відбувалося в інтегрованому середовищі розробки Android Studio мовою програмування Kotlin.

Задачею додатка буде підключення до поворотного столика через BLE. Отримання зображень об'єкту за допомогою вбудованої камери. Відправка команди повороту столика після кожного зробленого зображення. Після отримання зображень з усіх сторін об'єкту побудова тривимірної реконструкції.

Розглянемо алгоритм роботи додатка на рисунку 5.2. Алгоритм показано в дуже спрощеному види оскільки роботу додатка на операційній системі Android неможливо описати лінійним алгоритмом.

На початку роботи смартфон не під'єднано до обертового столика, тому необхідно їх з'єднати. Це робиться в автоматичному режимі, смартфон сам знаходить обертовий столик за MAC адресом Bluetooth.

Після підключення та виставлення смартфона та столика можна починати сканування. Воно починається після натискання кнопки «Capture». Після чого додаток в автоматичному режимі робитиме та зберігатиме зображення та управлятиме столиком. Всього сканер робить 72 зображення, обертаючи платформу на 5° відповідно.

Після закінчення сканування отримані знімки можна використати для ручної реконструкції методом фотограмметрії, або відправити їх на сервіс Reality Capture що надає компанії Autodesk.

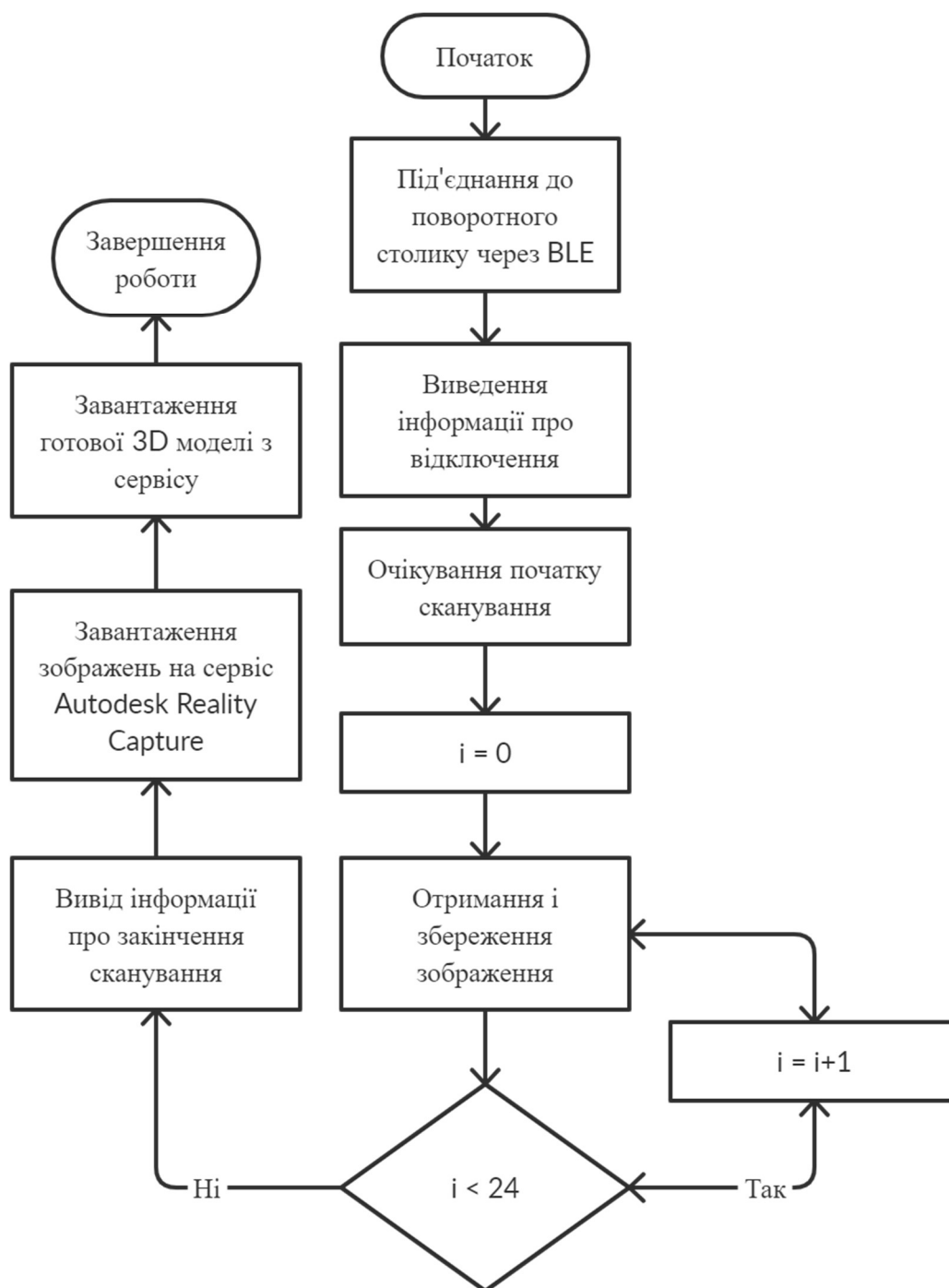


Рисунок 5.2 – Алгоритм роботи додатка для смартфона

### 5.3 Програмне забезпечення для реконструкції тривимірних зображень

Існує багато різного програмного забезпечення для реконструкції тривимірного зображення з набору знімків об'єкту методом фотограмметрії. Деякі з них вільні, деякі пропри тарні, а деякі представляються як сервіс. Всі вони

мають свої переваги та недоліки. Розглянемо деякі з доступних програмних забезпечень.

### 5.3.1 Meshroom

Meshroom представляє собою програмне забезпечення з відкритим сирцевим кодом, що використовує фреймворк AliceVision [24]. Фреймворк AliceVision з'явився в результаті співпраці великих європейських наукових та промислових партнерів, серед яких Чеський Технічний Університет, Національний Політехнічний Інститут Тулузи.

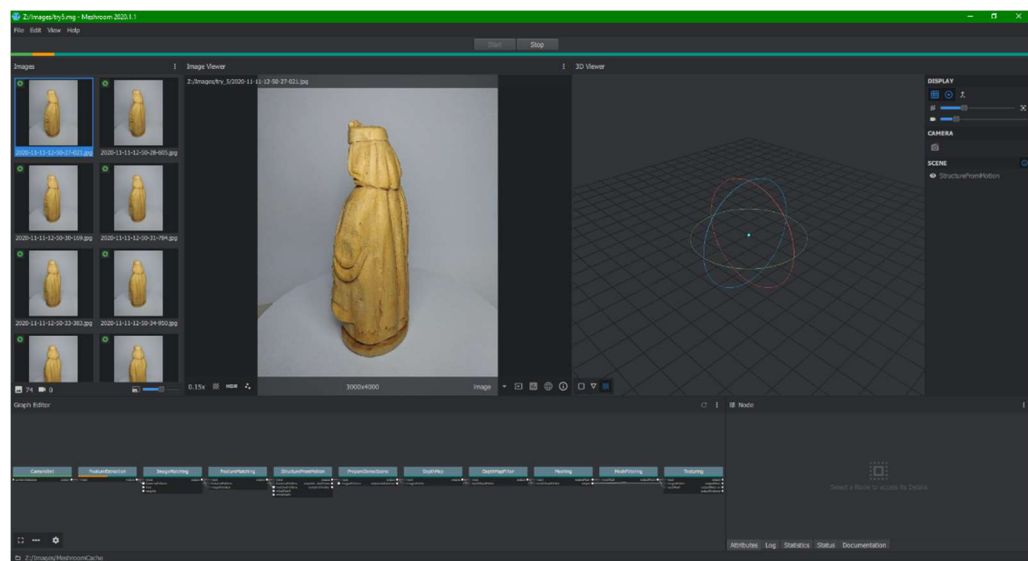


Рисунок 5.3 – Графічний інтерфейс ПЗ Meshroom

Meshroom дозволяє завантажити зображення об'єкту з різних сторін, натиснути кнопку пуск, та через деякий час отримати тривимірну модель, яку після деякої ручної обробки можна використовувати в тривимірному просторі, або надрукувати ти 3D принтері.

Обчислення тривимірної моделі відбуваються через «конвеєр», тобто обчислення виконуються одне за одним. Стандартний конвеєр фотограмметрії складається з двох основних частин:

- SfM - Structure-from-Motion (Структура з руху)
- MVS - MultiView-Stereo (Багато видова стереоскопія)

SfM робить висновок про структуру сцени з позицією та орієнтацією камер, результатом є набір віртуальних камер з розрідженою хмарою точок. Це видно на рисунку 5.4.

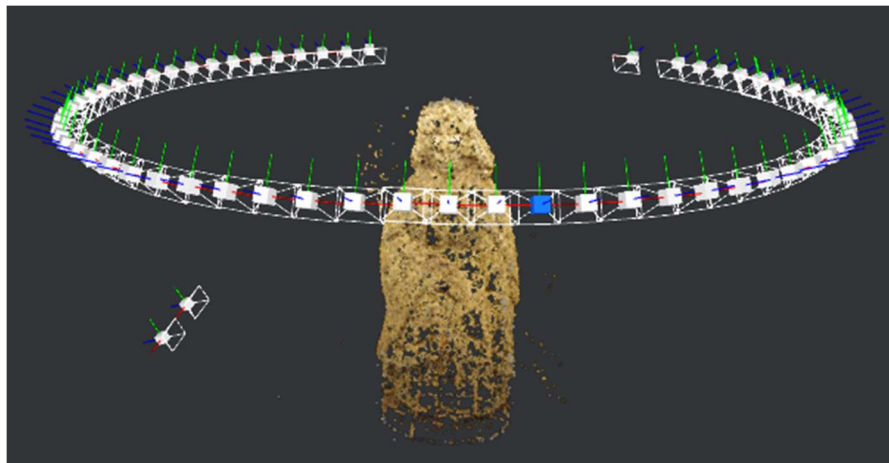


Рисунок 5.4 – Віртуальні камери розставлені навколо хмари точок

MVS використовує дані з кроку SfM щоб згенерувати цільну геометричну поверхню. Кінцевим результатом є файл формату OBJ з тривимірною текстурованою моделлю.

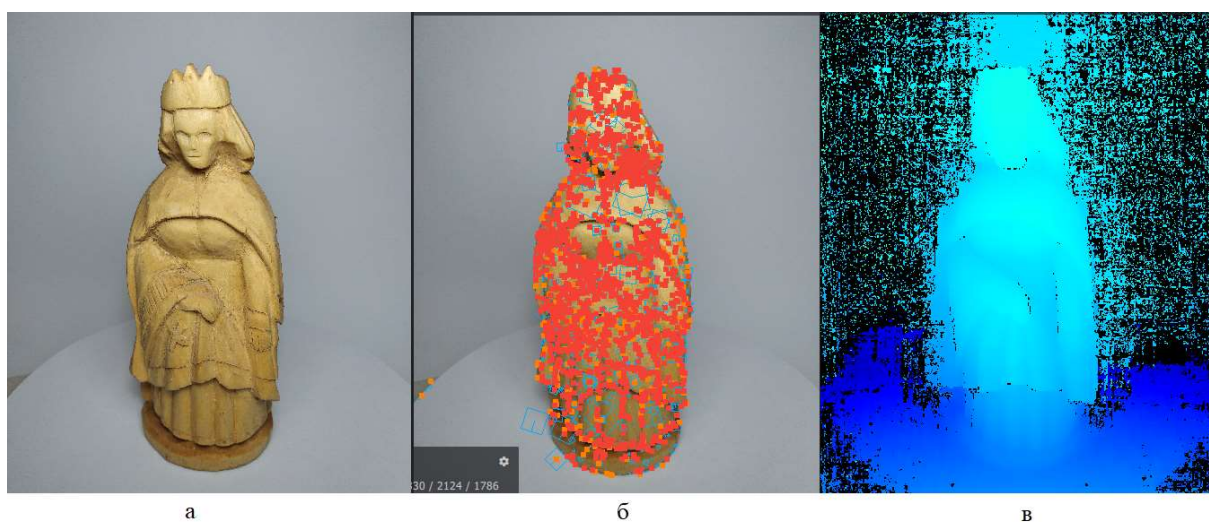


Рисунок 5.5 – Стадії обробки зображення

На рисунку 5.5 показані стадії обробки зображення програмним забезпеченням Meshroom. Початкове зображення показано на рисунку 5.5 а. З нього ПЗ отримує дані про камеру: фокусну відстань, розмір об'єктиву, параметр ISO, та деякі інші. На рисунку 5.5 б на початкове зображення накладено

знайдені виразні риси, зрівнюючи їх з рисами на інших зображеннях можна знайти положення камери, з якого було зроблено знімок. Після цього спеціальним алгоритмом знаходиться карта глибин на зображенні, її видно на рисунку 5.5 в. Чим ближче область на рисунку тим світліше відтінок.

На рисунку 5.6 показані стадії розрахунку тривимірної моделі. На рисунку 5.6 а показано каркасну модель з набору точок з'єднаних в трикутники, так звана полігональна сітка. З полігональної сітки робиться тіло, а якщо наспроєктувати текстуру то отримаємо віртуальне представлення реального тіла. Тверде тіло і тіло з текстурою зображені на рисунку 5.6 б і в відповідно.



Рисунок 5.6 – Готова тривимірна модель

Можна зробити висновок про переваги та недоліки програмного забезпечення Meshroom. Основною перевагою є те, що це відкрите програмне забезпечення, воно є безкоштовним та його можна налаштувати під конкретний проект. Недоліком є те, що для роботи необхідно мати персональний комп'ютер з графічним процесором Nvidia, що не завжди зручно.

### 5.3.2 Rescap pro

Rescap pro це програмне забезпечення для створення тривимірних моделей методом фотограмметрії. Його розробляє компанія Autodesk, відома по таким системам проектування як AutoCAD та Fusion 360.

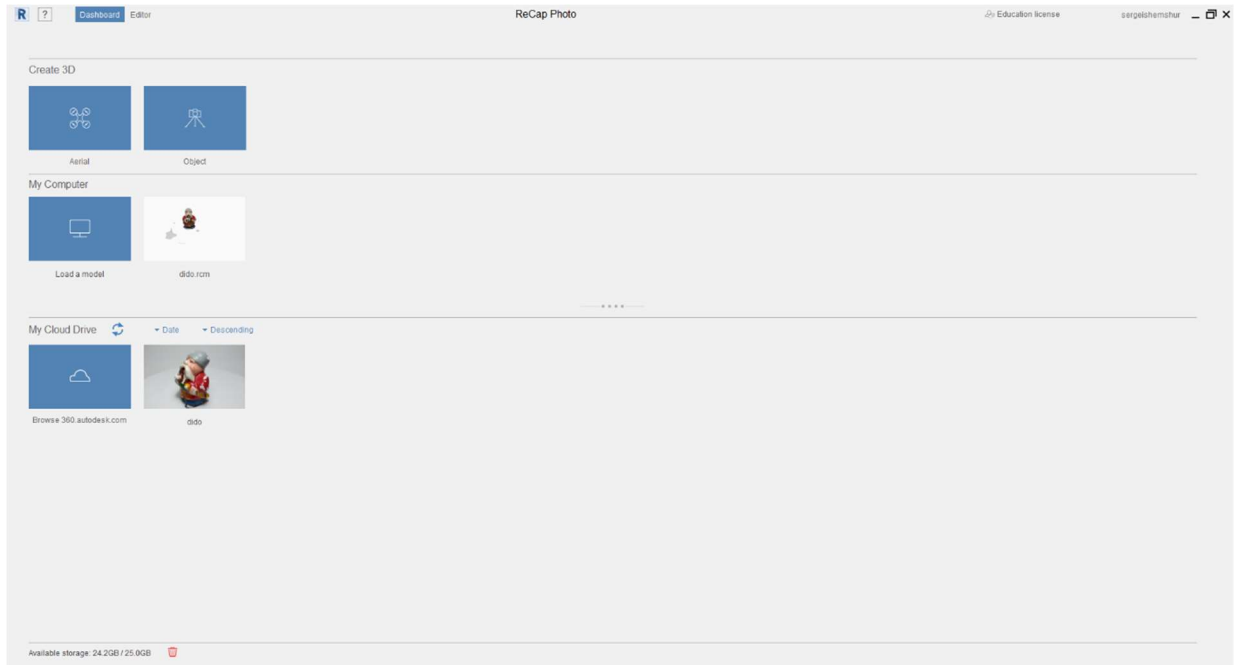


Рисунок 5.7 – Графічний інтерфейс ПЗ Rescap pro

На відміну від Meshroom Rescap pro має простий інтерфейс і функціонал. Програма на персональному комп'ютері являє собою інтерфейс для завантаження зображень на сервер та отримання уже готових тривимірних моделей з нього. Не потрібно налаштування, все працює без втручання користувача. Після обробки тривимірну модель можна завантажити та редагувати.

Також можливе завантаження зображень напряму на сервер, без персонального комп'ютера, як це було зроблено в додатку на Android. Autodesk надає прикладний програмний інтерфейс для цього.

Головною перевагою Autodesk Rescap pro є те, що обчислення проводяться на віддаленому сервері, що дозволяє не навантажувати машину користувача. Але недоліком є те, що цей сервіс є платним, хоча доступні версії для студентів.

## 6 РОЗРОБЛЕННЯ СТАРТАП ПРОЕКТУ

В цій частині буде проведено маркетинговий аналіз стартап проекту задля можливостей його ринкового впровадження та можливих напрямків реалізації [25].

### 6.1 Опис ідеї проекту

В цьому підпункті буде проведено аналіз змісту ідеї, напрямки застосування, основні вигоди для користувача товару та відмінності від аналогів (таблиця 6.1).

*Таблиця 6.1 Опис ідеї стартап проекту*

<i>Зміст ідеї</i>	<i>Напрямки застосування</i>	<i>Вигоди для користувача</i>
Розробка доступного 3D сканера	2. Медіа	1. Перенесення об'єктів в тривимірний простір. 2. Виробництво відеопродукції, графіки, реклами.
	3. Дизайн	1. Попередні пункти.
	4. Освіта	1. Можливість використання для навчання дітей в гуртках.
	4. Архітектура	1. Попередні пункти. 2. Можливість використання для реставрації малогабаритних об'єктів. 3. невеликі розміри дозволяють використовувати на об'єкті.

Проведемо аналіз потенційних техніко-економічних переваг ідеї в порівнянні з пропозиціями конкурентів (таблиця 6.2).

Таблиця 6.2 Визначення характеристик ідеї проекту

№ n/ n	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабка сторона)	N (нейтральна сторона)	S (сильна сторона)
		Мій проект	Matter and Form	HP	Shining 3D			
1	Простота	+	+	-	-			✓
2	Дешевизна	+	+	-	-			✓
3	Розміри капіталовкладень	+	-	-	-			✓

### 6.2 Технологічний аудит ідеї проекту

В цьому підрозділі буде проведено аудит технології, за допомогою якої можна реалізувати ідею проекту. В таблиці 6.3 зазначені технологічна здійсненність ідеї проекту.

Таблиця 6.3 Технологічна здійсненність ідеї проекту

№ n/n	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Розробка доступного 3D сканера	Виготовлення друкованих плат. Друк корпусу на 3D принтері.	Наявні	Доступні
2	Розробка програмного забезпечення для 3D сканера	Розробка додатків під систему Android	Наявні	Доступні
Обрана технологія реалізації ідеї проекту: замовлення виготовлення друкованих плат та монтажу компонентів, виготовлення корпусів на 3D принтерах. Постійне доопрацювання та додавання нових можливостей в ПЗ				

Всі технології виготовлення доступні, та дозволяють швидко створити невелику партію приладів власноруч.

### 6.3 Аналіз ризикових можливостей

Визначимо ринкові можливості, які можна використати під час ринкового впровадження проекту. Будуть розглянуті ринкові загрози що можуть перешкоджати реалізації проекту.

Було проведено аналіз попиту, результати якого наведено в таблиці 6.4.

Таблиця 6.4 Попередня характеристика ринку стартап-проекту

№ n/n	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	10
2	Загальний обсяг продаж, грн/ум.од	Невідомо
3	Динаміка ринку (якісна оцінка)	Зростає
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	50%

За попередніми оцінками можна зробити висновки що в даний час ринок є зростаючим та привабливим для входження.

Далі буде визначено, будуть дані їх характеристики та будуть сформований орієнтовний перелік вимог до товару.

Таблиця 6.5 Характеристика потенційних клієнтів стартап проекту

№ n/n	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Потреба в 3D сканері	Користувачі 3D принтерів	Пошук дешевих 3D сканерів	–Невелика ціна
2		3D художники	Пошук сканерів, що можуть дати якісний результат	–Надійність –Якість 3D сканів

Далі визначимо фактори, що перешкоджають ринковому впровадженні.

Таблиця 6.6 Фактори загроз

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст загрози</i>	<i>Можлива реакція компанії</i>
1	Поява нових, дешевших аналогів	Поява нових технологій, що дозволяють робити те ж саме, але дешевше	Втрата клієнтів
2	Фінансова криза	Зниження покупної спроможності споживачів	Втрата клієнтів Вимушене зниження цін Спад попиту

Розглянемо фактори що сприяють ринковому впровадженні проекту.

Таблиця 6.7 Фактори можливостей

<i>№ n/n</i>	<i>Фактор</i>	<i>Зміст можливості</i>	<i>Можлива реакція компанії</i>
1	Зростання доходу населення	Збільшення кількості споживачів	Збільшення прибутків, зріст обсягів продажів
2	Поява дешевших технологій виробництва	Зменшення собівартості виробництва	Попередні пункти
3	Ослаблення фірм конкурентів	Перехід споживачів від конкурентів	Попередні пункти

Проведемо аналіз загальних рис конкуренції на ринку.

Таблиця 6.8 Ступеневий аналіз конкуренції на ринку

<i>Особливості конкурентного середовища</i>	<i>В чому проявляється дана характеристика</i>	<i>Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)</i>
1. Тип конкуренції: монополістична	На ринку є багато незалежних конкуруючих фірм.	Необхідно розробити одну продукцію, що зможе конкурувати з наявною на ринку
2. За рівнем конкурентної боротьби: міжнародна	На ринку представлена продукція від виробників з різних країн світу	Вихід на міжнародний ринок
3. За галузевою ознакою: внутрішньогалузева	Боротьба між виробниками в одній галузі виробництва. За зниження затрат, підвищення якості	Збільшення якості продукції

Таблиця 6.8 (Продовження)

4. Конкуренція за видами товарів: товарно-видова	Споживачі вибирають між товарами одного виду від різних виробників	Зниження витрат на виробництво
5. За характером конкурентних переваг: цінова	Боротьба через зниження цін на товари шляхом зниження ціни виробництва	Зниження витрат на виробництво
6. За інтенсивністю: марочна	Компанії пропонують різний товар за різними цінами	Вибір ніші, та націлення на неї

Проведемо аналіз конкуренції в галузі за моделлю п'ять сил Майкла Портера (таблиця 6.9)

Таблиця 6.9 Аналіз конкуренції за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	<i>Mirobo Atlas</i>	<i>HP, Matter and Form</i>	Визначити фактори сили постачальників	Визначити фактори сили споживачів	Фактори загроз з боку замінників
Висновки:	Визначити інтенсивність конкурентної боротьби з боку прямих конкурентів	Вихід на ринок можливий	Чи постачальники диктують умови роботи на ринку?  Постачальники не диктують умови	Чи клієнти диктують умови роботи на ринку?  Умови клієнтів: невисока ціна, гарна роздільна здатність.	Обмеження для роботи на ринку через товари-замінники

На основі аналізу конкуренції визначимо та обґрунтуємо перелік факторів конкурентоспроможності(таблиця 6.10).

Таблиця 6.10 Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування
1	Ціна	Використання смартфона для побудови зображень зменшує ціну продукту

Таблиця 6.10 (Продовження)

2	Простота	Не потрібне довге налаштування, сканування однією кнопкою
3	Рухомість	Можливість роботи від акумулятора

За факторами конкурентоспроможності проведемо аналіз сильних та слабких сторін проекту.

Таблиця 6.11 Порівняльний аналіз сильних та слабких сторін 3D сканера

№ п/ п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з ... (назва підприємства)						
			-3	-2	-1	0	+1	+2	+3
1	Ціна	20	17						
2	Простота	10			9				
3	Рухомість	10		8					

Проведемо SWOT аналіз проекту(таблиця 6.12).

Таблиця 6.12 SWOT-аналіз стартап-проекту

Сильні сторони: Низька ціна Простота роботи	Слабкі сторони: Відсутність стартового капіталу
Можливості: Введення нових технологій	Загрози: Зменшення покупної спроможності потенційного покупця

Після проведення SWOT аналізу доцільно розробити альтернативи ринкової поведінки.

Таблиця 6.13 Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Реклама в мережі Інтернет	Висока	3 – 4 місяці
2	Продаж не готового пристрою, а набору для збирання.	Середня	5 – 7 місяців
3	Використання платформ для стартапів (Kickstarter)	Висока	1 – 6 місяців

Після огляду альтернатив ринкового впровадження стартап проекту було обрано третій варіант альтернативи, використання платформ для стартапів по типу Kickstarter

#### 6.4 Розроблення ринкової стратегії проекту

Для розроблення ринкової стратегії спочатку визначимо стратегії охоплення ринку, опишемо цільові групи потенційних споживачів(таблиця 6.14).

Таблиця 6.14 Вибір цільових груп потенційних споживачів

№ n/n	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Аматори 3D друку	Висока	Високий	Низька	Відносно просто
2	Професійні користувачі	Низька	Високий	Висока	Важко
Які цільові групи обрано: було обрано аматорську цільову групу, оскільки професіоналам необхідно якісний пристрій, при цьому ціна відходить на третій план, а в аматорів ціна на першому місці.					

Виберемо стратегію охоплення ринку(таблиця 6.15)

Таблиця 6.15 Визначення базової стратегії розвитку

№ n/n	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Концентрований маркетинг	Низька ціна, простота	Набагато менша ціна	Спеціалізація

Виберемо стратегію конкурентної поведінки(таблиця 6.16)

Таблиця 6.16 Визначення базової конкурентної поведінки

№ n/n	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Ні	Шукати нових	Ні	Стратегія лідера

Розробимо стратегію позиціонування(таблиця 6.17)

Таблиця 6.17 Визначення стратегії позиціонування

№ n/ n	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувавши комплексну позицію власного проекту (три ключових)
1	Якість Невелика ціна Надійність	Використання інтернету для реклами	Низька ціна Використання камери смартфона для отримання знімків	Дешевий Простий Надійний

### 6.5 Розроблення маркетингової програми стартап-проекту

Сформуємо маркетингову концепцію товару який отримає споживач. В таблиці 6.18 підсумовано результати попереднього аналізу конкурентоспроможності товару.

Таблиця 6.18 Визначення ключових переваг конкретності потенційного товару

№ n/n	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Низька ціна	Невелика ціна	Дешевий у виготовленні
	Простота керування	Просте керування	Пристрій є простим в керуванні
	Габарити	Малі габарити	Дозволяє просте перенесення
	Роздільна здатність 3D сканів	Висока роздільна здатність сканування	

Розробимо трирівнева маркетингову модель товару(таблиця 6.19).

Таблиця 6.19 Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Опис базової потреби споживача, яку задовольняє товар (згідно концепції), її основної функціональної вигоди		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1. ....		
	2. ....		
	Якість: стандарти, нормативи, параметри тестування тощо		
Пакування			

Таблиця 6.19 (Продовження)

	Марка: назва організації-розробника + назва товару
III. Товар із підкріпленням	До продажу
	Після продажу
За рахунок чого потенційний товар буде захищено від копіювання: за рахунок оформлення міжнародного патенту.	

Визначимо цінові межі, якими необхідно керуватися при встановленні цін на потенційний товар.

Таблиця 6.20 Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	3000 - 4000	5000 - 50000	Середній	2000 - 5000

Визначимо оптимальну систему збуту

Таблиця 6.21 Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Закупівля найдешевшого приладдя	Вільний доступ до товару	Невідомо	Збут через мережу інтернет

Проведемо розробку концепцію маркетингових комунікацій(таблиця 6.22)

Таблиця 6.22 Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Невідомо	Інтернет	Можливості пристрою	Донести до потенційного споживача можливості продукту	Донести сильні сторони продукту.

## ВИСНОВОК

Після проведення маркетингового дослідження можна зробити висновок що розроблений продукт має місце на ринку. Він привабливий для певної групи споживачів. Є перспективи впровадження проекту, хоча подальше введення продукту на ринок можливо лише при знаходженні інвесторів.

## ВИСНОВКИ

- 1) Було проаналізовано різні методи реконструкції тривимірного зображення. Було визначено їх переваги та недоліки. Після аналізу було обрано метод фотограмметрії, через свою простоту відносно інших.
- 2) Було розглянуто аналогічні пристрої. Після їх розгляду було прийнято рішення використовувати схожу конструкцію з обертовим столиком. Також було висунуто основні вимоги до конструкції.
- 3) Було проаналізовано вибір елементної бази компонентів. Після чого було розроблено пристрій управління поворотним столиком. Проектування відбувалося в системі автоматизованого проектування радіоелектронних засобів Altium Designer.
- 4) Створено комплект конструкторської документації, що відповідає усім вимогам ЄСКД. Також було створено робочий прототип пристрою.
- 5) Після проведення експериментальних досліджень була досягнута основна ціль магістерської дисертації – можливість створювати тривимірні моделі реальних об'єктів.

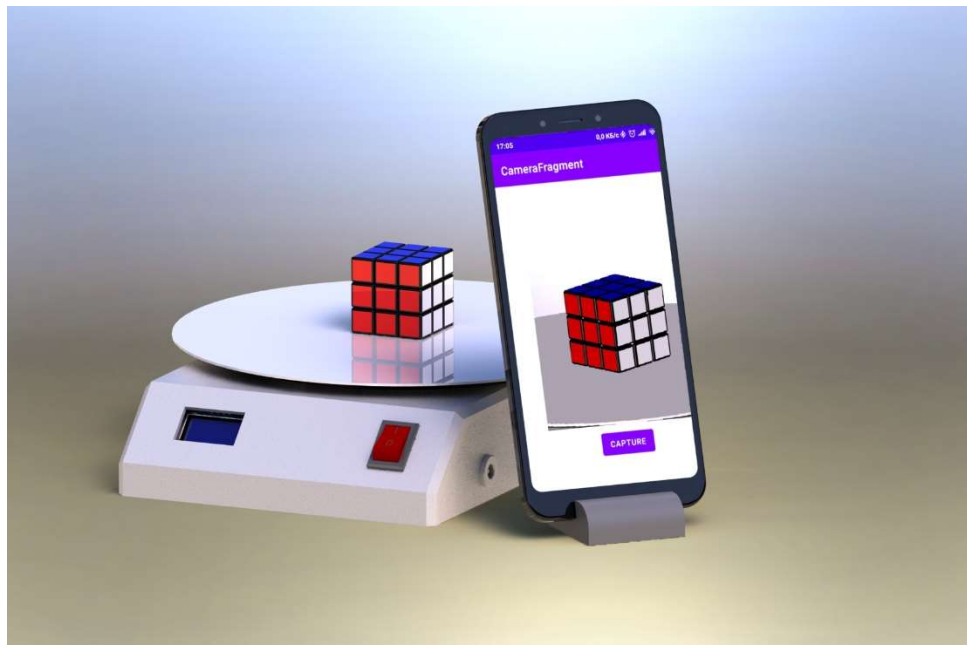


Рисунок 1 – Спроектований пристрій

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

- 1) An Introduction of 3D Scanning and 3D scanners [Електронний ресурс]. Режим доступу: <https://www.revopoint3d.com/an-introduction-of-3d-scanning-and-3d-scanners/> – Назва з екрану
- 2) 3D shape scanning with a time-of-flight camera [Електронний ресурс]. Режим доступу: <https://www.semanticscholar.org/paper/3D-shape-scanning-with-a-time-of-flight-camera-Cui-Schuon/a55c83982dc0fbea23cbabdce1d5ddfa3591ab5c> – Назва з екрану
- 3) Understanding laser-based 3D triangulation methods [Електронний ресурс]. Режим доступу: <https://www.vision-systems.com/cameras-accessories/article/16738248/understanding-laserbased-3d-triangulation-methods> – Назва з екрану
- 4) Structured Light 3D Scanning [Електронний ресурс]. Режим доступу: <https://www.instructables.com/Structured-Light-3D-Scanning/> – Назва з екрану
- 5) Geodetic Systems [Електронний ресурс]. Режим доступу: <https://www.geodetic.com/> – Назва з екрану
- 6) Reality capture software [Електронний ресурс]. Режим доступу: <https://www.autodesk.com/products/recap/overview> – Назва з екрану
- 7) 3DF ZEPHYR The Complete Photogrammetry Solution [Електронний ресурс]. Режим доступу: <https://www.3dflow.net> – Назва з екрану
- 8) ESP32-WROOM-32 Datasheet [Електронний ресурс]. Режим доступу: [https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf) – Назва з екрану
- 9) MicroPython [Електронний ресурс]. Режим доступу: <http://micropython.org> – Назва з екрану
- 10) Small Reduction Stepper Motor - 12VDC 32-Step 1/16 Gearing [Електронний ресурс]. Режим доступу: <https://www.adafruit.com/product/918> – Назва з екрану

- 11) Can the output current of the esp32 GPIO be reduced? [Электронный ресурс]. Режим доступа: <https://esp32.com/viewtopic.php?f=2&t=2027> – Назва з екрану
- 12) ULN2003A 50-V, 7-ch darlington transistor array [Электронный ресурс]. Режим доступа: <https://www.ti.com/product/ULN2003A?HQS=TI-null-null-alldatasheets-df-pf-SEP-wwe&DCM=yes&dclid=CM247qyUq-0CFYMGiwodS10B1A#product-details###params> – Назва з екрану
- 13) 128 x 64 Dot Matrix OLED/PLED Segment/Common Driver with Controller [Электронный ресурс]. Режим доступа: <https://cdn-shop.adafruit.com/datasheets/SSD1306.pdf> – Назва з екрану
- 14) Блок питания, сетевой адаптер 12В 3А, 5.5x2.1мм 5.5x2.5мм [Электронный ресурс]. Режим доступа: <https://panas.com.ua/p1126081761-blok-pitaniya-setevoj.html> – Назва з екрану
- 15) Interface OLED Graphic Display Module with Arduino [Электронный ресурс]. Режим доступа: <https://lastminuteengineers.com/oled-display-arduino-tutorial/> – Назва з екрану
- 16) AM1S-Z DC-DC Converter [Электронный ресурс]. Режим доступа: <https://www.tme.eu/Document/b5fea7ee234bf8823f018913de2600f1/AM1S-Z.pdf> – Назва з екрану
- 17) XH connector [Электронный ресурс]. Режим доступа: [http://www.jst-mfg.com/product/detail\\_e.php?series=277](http://www.jst-mfg.com/product/detail_e.php?series=277) – Назва з екрану
- 18) CGTrader - 3D Models for VR / AR and CG projects | CGTrader [Электронный ресурс]. Режим доступа: <https://www.cgtrader.com/>. – Назва з екрану
- 19) Polymer Data Handbook (2nd Edition) / James E. Mark., Oxford University Press, 2009.
- 20) ESP-IDF Programming Guide [Электронный ресурс]. Режим доступа: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32> – Назва з екрану
- 21) Firmware for Generic ESP32 module [Электронный ресурс]. Режим доступа: <http://micropython.org/download/esp32/> – Назва з екрану

22) How Bluetooth Low Energy Works [Електронний ресурс]. Режим доступу: <https://www.novelbits.io/bluetooth-low-energy-advertisements-part-1/> –

Назва з екрану

23) UART/Serial Port Emulation over BLE [Електронний ресурс]. Режим доступу:

[https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.sdk5.v14.0.0%2Fble\\_sdk\\_app\\_nus\\_eval.html](https://infocenter.nordicsemi.com/index.jsp?topic=%2Fcom.nordic.infocenter.sdk5.v14.0.0%2Fble_sdk_app_nus_eval.html) – Назва з екрану

24) AliceVision | MeshRoom [Електронний ресурс]. Режим доступу: <https://alicevision.org/#meshroom> – Назва з екрану

25) Розроблення стартап-проекту / За заг. ред. О.А. Гавриша. – Київ: НТУУ «КПІ», 2016. – 28 с.

## ДОДАТОК А

## Лістинг програмного коду

Файл **main.py**

```
from BLE import BLEPeripheral
from ssd1306 import SSD1306_I2C
from machine import I2C, Pin
from Stepper import Stepper
import time

# Oled setup
sdaPin = Pin(21, Pin.OUT)
sclPin = Pin(22, Pin.OUT)

i2c = I2C(scl=sclPin, sda=sdaPin)
oled = SSD1306_I2C(128, 64, i2c)

oled.text("3D Scanner", 0, 56)
oled.show()

def logToOled(string):
    oled.fill(0x0000)
    oled.text(string, 0, 28)
    oled.show()

# BLE setup
def bleOnWriteCallback(data):
    data = data.decode("utf-8")
    data = removeSpecialCharacters(data)
    print(data)
    logToOled(data)
    doAction(data)
    ble.send("OK")
```

```

ble = BLEPeripheral(name="Scanner")
ble.on_write(bleOnWriteCallback)

# Stepper setup
stepperPins = [Pin(13, Pin.OUT), Pin(12, Pin.OUT), Pin(14, Pin.OUT), Pin(27, Pin.OUT)]

stepper = Stepper(
    "HALF_STEP", stepperPins[0], stepperPins[1], stepperPins[2], stepperPins[3], 1
)

# action consist of command and value
# example "TL-5" - turn left 5 degrees
# actions : TL - turn left
#           TR - turn right
def doAction(action: str):
    command, value = parseAction(action)

    print("doAction", action, command, value)

    if command == "TR":
        stepper.angle(value, direction=1)
    elif command == "TL":
        stepper.angle(value, direction=-1)
    else:
        raise Exception("Incorrect command")

def parseAction(action):
    action = action.split("-")
    return (action[0], int(action[1]))

def removeSpecialCharacters(string):
    specialChars = ["\n", "\r"]

```

```
buf = str()
for char in string:
    if char not in specialChars:
        buf += char
return buf
```

## Файл BLE.py

```
# Bluetooth LE peripheral

import struct
import bluetooth

from micropython import const

_IRQ_CENTRAL_CONNECT = const(1)
_IRQ_CENTRAL_DISCONNECT = const(2)
_IRQ_GATTS_WRITE = const(3)

_UART_UUID = bluetooth.UUID("6E400001-B5A3-F393-E0A9-E50E24DCCA9E")
_UART_TX = (
    bluetooth.UUID("6E400003-B5A3-F393-E0A9-E50E24DCCA9E"),
    bluetooth.FLAG_READ | bluetooth.FLAG_NOTIFY,
)
_UART_RX = (
    bluetooth.UUID("6E400002-B5A3-F393-E0A9-E50E24DCCA9E"),
    bluetooth.FLAG_WRITE | bluetooth.FLAG_WRITE_NO_RESPONSE,
)
_UART_SERVICE = (
    _UART_UUID,
    (_UART_TX, _UART_RX),
)

_ADV_TYPE_FLAGS = const(0x01)
_ADV_TYPE_NAME = const(0x09)
_ADV_TYPE_UUID16_COMPLETE = const(0x3)
_ADV_TYPE_UUID32_COMPLETE = const(0x5)
_ADV_TYPE_UUID128_COMPLETE = const(0x7)
_ADV_TYPE_UUID16_MORE = const(0x2)
_ADV_TYPE_UUID32_MORE = const(0x4)
_ADV_TYPE_UUID128_MORE = const(0x6)
```

```

_ADV_TYPE_APPEARANCE = const(0x19)

class BLEPeripheral:
    def __init__(self, name="mpy-uart"):
        self._ble = bluetooth.BLE()
        self._ble.active(True)
        self._ble.irq(self._irq)
        ((self._handle_tx, self._handle_rx),) = self._ble.gatts_register_services(
            (_UART_SERVICE,)
        )
        self._connections = set()
        self._write_callback = None
        self._payload = self.advertising_payload(name=name, services=[_UART_UUID])
        self._advertise()

    def _irq(self, event, data):
        # Track connections so we can send notifications.
        if event == _IRQ_CENTRAL_CONNECT:
            conn_handle, _, _ = data
            print("New connection", conn_handle)
            self._connections.add(conn_handle)
        elif event == _IRQ_CENTRAL_DISCONNECT:
            conn_handle, _, _ = data
            print("Disconnected", conn_handle)
            self._connections.remove(conn_handle)
            # Start advertising again to allow a new connection.
            self._advertise()
        elif event == _IRQ_GATTS_WRITE:
            conn_handle, value_handle = data
            value = self._ble.gatts_read(value_handle)
            if value_handle == self._handle_rx and self._write_callback:
                self._write_callback(value)

```

```

def advertising_payload(
    self, limited_disc=False, br_edr=False, name=None, services=None, appearance=0
):
    payload = bytearray()

    def _append(adv_type, value):
        nonlocal payload
        payload += struct.pack("BB", len(value) + 1, adv_type) + v
alue

    _append(
        _ADV_TYPE_FLAGS,
        struct.pack(
            "B", (0x01 if limited_disc else 0x02) + (0x18 if br_ed
r else 0x04)
        ),
    )

    if name:
        _append(_ADV_TYPE_NAME, name)

    if services:
        for uuid in services:
            b = bytes(uuid)
            if len(b) == 2:
                _append(_ADV_TYPE_UUID16_COMPLETE, b)
            elif len(b) == 4:
                _append(_ADV_TYPE_UUID32_COMPLETE, b)
            elif len(b) == 16:
                _append(_ADV_TYPE_UUID128_COMPLETE, b)

```

```
# See org.bluetooth.characteristic.gap.appearance.xml
if appearance:
    _append(_ADV_TYPE_APPEARANCE, struct.pack("<h", appearance
))

# print(payload)
return payload

def send(self, data):
    for conn_handle in self._connections:
        self._ble.gatts_notify(conn_handle, self._handle_tx, data)

def is_connected(self):
    return len(self._connections) > 0

def _advertise(self, interval_us=500000):
    print("Starting advertising")
    self._ble.gap_advertise(interval_us, adv_data=self._payload)

def on_write(self, callback):
    self._write_callback = callback
```

## Файл `ssd1306.py`

```
# MicroPython SSD1306 OLED driver, I2C and SPI interfaces

from micropython import const
import framebuf

# register definitions
SET_CONTRAST = const(0x81)
SET_ENTIRE_ON = const(0xA4)
SET_NORM_INV = const(0xA6)
SET_DISP = const(0xAE)
SET_MEM_ADDR = const(0x20)
SET_COL_ADDR = const(0x21)
SET_PAGE_ADDR = const(0x22)
SET_DISP_START_LINE = const(0x40)
SET_SEG_REMAP = const(0xA0)
SET_MUX_RATIO = const(0xA8)
SET_COM_OUT_DIR = const(0xC0)
SET_DISP_OFFSET = const(0xD3)
SET_COM_PIN_CFG = const(0xDA)
SET_DISP_CLK_DIV = const(0xD5)
SET_PRECHARGE = const(0xD9)
SET_VCOM_DESEL = const(0xDB)
SET_CHARGE_PUMP = const(0x8D)

# Subclassing FrameBuffer provides support for graphics primitives
# http://docs.micropython.org/en/latest/pyboard/library/framebuf.html
class SSD1306(framebuf.FrameBuffer):
    def __init__(self, width, height, external_vcc):
        self.width = width
        self.height = height
        self.external_vcc = external_vcc
        self.pages = self.height // 8
```

```

        self.buffer = bytearray(self.pages * self.width)
        super().__init__(self.buffer, self.width, self.height, framebuffer.MONO_VLSB)
        self.init_display()

# Abstract
def write_data(self, buf):
    pass

# Abstract
def write_cmd(self, cmd):
    pass

def init_display(self):
    for cmd in (
        SET_DISP | 0x00, # off
        # address setting
        SET_MEM_ADDR,
        0x00, # horizontal
        # resolution and layout
        SET_DISP_START_LINE | 0x00,
        SET_SEG_REMAP | 0x01, # column addr 127 mapped to SEG0
        SET_MUX_RATIO,
        self.height - 1,
        SET_COM_OUT_DIR | 0x08, # scan from COM[N] to COM0
        SET_DISP_OFFSET,
        0x00,
        SET_COM_PIN_CFG,
        0x02 if self.width > 2 * self.height else 0x12,
        # timing and driving scheme
        SET_DISP_CLK_DIV,
        0x80,
        SET_PRECHARGE,
        0x22 if self.external_vcc else 0xF1,

```

```

        SET_VCOM_DESEL,
        0x30, # 0.83*Vcc
        # display
        SET_CONTRAST,
        0xFF, # maximum
        SET_ENTIRE_ON, # output follows RAM contents
        SET_NORM_INV, # not inverted
        # charge pump
        SET_CHARGE_PUMP,
        0x10 if self.external_vcc else 0x14,
        SET_DISP | 0x01,
    ): # on
        self.write_cmd(cmd)
    self.fill(0)
    self.show()

def poweroff(self):
    self.write_cmd(SET_DISP | 0x00)

def poweron(self):
    self.write_cmd(SET_DISP | 0x01)

def contrast(self, contrast):
    self.write_cmd(SET_CONTRAST)
    self.write_cmd(contrast)

def invert(self, invert):
    self.write_cmd(SET_NORM_INV | (invert & 1))

def show(self):
    x0 = 0
    x1 = self.width - 1
    if self.width == 64:
        # displays with width of 64 pixels are shifted by 32

```

```
        x0 += 32
        x1 += 32
self.write_cmd(SET_COL_ADDR)
self.write_cmd(x0)
self.write_cmd(x1)
self.write_cmd(SET_PAGE_ADDR)
self.write_cmd(0)
self.write_cmd(self.pages - 1)
self.write_data(self.buffer)
```

```
class SSD1306_I2C(SSD1306):
    def __init__(self, width, height, i2c, addr=0x3C, external_vcc=False):
        self.i2c = i2c
        self.addr = addr
        self.temp = bytearray(2)
        self.write_list = [b"\x40", None] # Co=0, D/C#=1
        super().__init__(width, height, external_vcc)

    def write_cmd(self, cmd):
        self.temp[0] = 0x80 # Co=1, D/C#=0
        self.temp[1] = cmd
        self.i2c.writeto(self.addr, self.temp)

    def write_data(self, buf):
        self.write_list[1] = buf
        self.i2c.writevto(self.addr, self.write_list)
```

## Файл `Stepper.py`

```

import time

# only test for uln2003

class Stepper:
    FULL_ROTATION = 4075.7728395061727 / 8
    # http://www.jangeox.be/2013/10/stepper-motor-28byj-48_25.html

    HALF_STEP = [
        [0, 0, 0, 1],
        [0, 0, 1, 1],
        [0, 0, 1, 0],
        [0, 1, 1, 0],
        [0, 1, 0, 0],
        [1, 1, 0, 0],
        [1, 0, 0, 0],
        [1, 0, 0, 1],
    ]

    FULL_STEP = [[1, 0, 1, 0], [0, 1, 1, 0], [0, 1, 0, 1], [1, 0, 0, 1]
]]

    def __init__(self, mode, pin1, pin2, pin3, pin4, delay):
        if mode == "FULL_STEP":
            self.mode = self.FULL_STEP
        else:
            self.mode = self.HALF_STEP

        self.pin1 = pin1
        self.pin2 = pin2
        self.pin3 = pin3
        self.pin4 = pin4

        self.delay = delay # Recommend 10+ for FULL_STEP, 1 is OK for
HALF_STEP

        self.reduction = 1 / 7

```

```

# Initialize all to 0
self.reset()

def step(self, count, direction=1):
    """Rotate count steps. direction = -1 means backwards"""
    for x in range(count):
        for bit in self.mode[::-1]:
            self.pin1(bit[0])
            self.pin2(bit[1])
            self.pin3(bit[2])
            self.pin4(bit[3])
            time.sleep_ms(self.delay)
        self.reset()

def angle(self, r, direction=1):
    self.step(int((self.FULL_ROTATION / self.reduction) * r / 360)
, direction)

def reset(self):
    # Reset to 0, no holding, these are geared, you can't move the
m
    self.pin1(0)
    self.pin2(0)
    self.pin3(0)
    self.pin4(0)

def create(pin1, pin2, pin3, pin4, delay=2, mode="HALF_STEP"):
    return Stepper(mode, pin1, pin2, pin3, pin4, delay)

```

## ДОДАТОК Б

### Перелік елементів пристрою



Поз. обозначение	Наименование	Кол.	Примечание
R1	Резистор RCS1206100KFKEA	1	Ф. Vishay / Dale
S1	Кнопка IT-1187-160GM-G	1	Ф. Switronic
U1	Мікроконтролер ESP32-WROOM-32D	1	Ф. Espressif Systems
U2	Транзисторна збірка ULN2003D	1	Ф. Texas Instruments
XP1	Вилка 77311-4AC-03LF	1	Ф. Amphenol ICC
XP2	Вилка B4B-XH-A	1	Ф. JST
XP3	Вилка B5B-XH-A	1	Ф. JST
XT1	Клемник DG301-5.0-02P-12-00A(H)	1	Ф. Degson
A4	<u>Індикатор SSD1306</u>	1	

Інв. № публ.	Взам. инв. №	Полп. и. лая	Полп. И. лая	Изм.	Лист	№ докум.	Подп.	Дата	РА-мп91.468469.001 ПЭЗ	Лист
										76

## ДОДАТОК В

### Специфікація на пристрій управління

Перв. підмен		Формат	Зона	Поз.	Обозначение	Наименование	Кол.	Примечание	
Справ. №		000000							
АБВГ.123456.799									
A2		РА-мп91.468469.001ЭЗ							
A4		РА-мп91.468469.001 ПЭЗ							
		<u>Документація</u>							
		<u>Деталі</u>							
A3		1	РА-мп91.785715.001				Плата друкована	1	
		<u>Інші вироби</u>							
Попл. и дата		08.08.08		2	Конденсатор ТАJD107M010RNJ		1	С1	
				3	Конденсатор СС1206KRX7R9BB104		3	С2...С3	
Изм. № публ.		000000		4	DC – DC перетворювач AM1S-1203SH30Z		1	D1	
Взам. инв. №		000000		5	Резистор RCS1206100KFKEA		1	R1	
Попл. и дата		08.08.08							
Изм. № попл.		000000							
		Изм	Лист	№ докум.	Подп.	Дата			
		Разраб.	Шемшур						
		Пров.	Бондаренко						
		Нач.сект.							
		Н.контр.							
		Утв.							
		РА-мп91.468534.001							
		Пристрій управління					Лит.	Лист	Листов
							У	78	84



## ДОДАТОК Г

### Специфікація на сканер



