BANGLADESH UNIVERSITY OF BUSINESS AND TECHNOLOGY (BUBT)

REPORT On (“Tello – AI-Powered Smart Learning Toy with IoT Connectivity”)

Course code: CSE 426

Course title: Internet of Things Lab

Developed By:

Name	ID
Most. Sonia Islam	21225103101
Lamia Muntaha	21225103185
Md. Mehedi Hasan	21225103334
Abdullah Al Hill Baki Anim	21225103341
Humayra Kabir Hride	21225103501
Insana Rahman	21225103343

Supervised by:

Sayefa Arafah

Lecturer

Department of Computer Science and Engineering

Bangladesh University of Business and Technology (BUBT)

19 MAY , 2025

DECLARATION

We hereby declare that the project entitled " **Tello** – AI-Powered Smart Learning Toy with IoT Connectivity" submitted for the completion of **CSE 426: Internet of Things l**ab course in the faculty of Computer Science and Engineering of Bangladesh University of Business and Technology (BUBT), is our original work

Most. Sonia Islam	**Lamia Muntaha**	**Md. Mehedi Hasan**
ID: 21225103101	**ID: 21225103185**	**ID: 21225103334**
Intake: 49	**Intake: 49**	**Intake: 49**
Section-07	**Section-07**	**Section-07**

Abdullah Al Hill Baki Anim	**Insana Rahman**	**Humayra Kabir Hride**
ID: 21225103341	**ID: 21225103343**	**ID: 21225103501**
Intake: 49	**Intake: 49**	**Intake: 49**
Section-07	**Section-07**	**Section-07**

ACKNOWLEDGEMENTS

I would like to thank the following people for their help in the production of this project: **Sayefa Arafah** miss, ***Lecturer***, Dept. of **CSE**, BUBT, & our project supervisor for all of his ongoing assistance with the project, without whose help and support throughout, this project would not have been possible.

We should thank our entire associate at **Bangladesh University Of Business And Technology**, who took in this inspection while completing the class work. Finally, we ought to perceive with due respect the reliable assistance and patients of our parents.

ABSTRACT

The **Tello** – AI-Powered Smart Learning Toy with IoT Connectivity is an innovative AI-powered smart learning toy designed to make early education interactive, engaging, and technologically enriched. Utilizing the ESP32-CAM module, Tello detects and identifies real-world objects through real-time image capture and cloud-based machine learning models. The identified object names are displayed on a compact OLED screen and simultaneously converted to speech using the Google Text-to-Speech (TTS) API and a MAX98357A audio amplifier. This combination of visual and auditory feedback enhances children's learning experiences by catering to multiple sensory inputs. Wi-Fi connectivity ensures seamless access to cloud resources, enabling real-time updates and expanding the system's capabilities. Tello is designed to be lightweight, portable, and scalable, with potential future expansions including learning progress tracking, AI-powered Q&A, and parental control features. By integrating AI, IoT, and cloud services into a single educational tool, Tello stands out as a powerful aid for smart and effective early childhood learning.

APPROVAL

This report “Tello-AI-Powered Smart Learning Toy with IoT Connectivity” submitted by Most. Sonia Islam, Lamia Muntaha, Md. Mehedi Hasan, Humayra Kabir Hride, Insana Rahman, Abdullah Al Hill Baki Anim, students of the Department of Computer Science and Engineering (CSE), **Bangladesh University of Business and Technology (BUBT)**, has been carried out under the supervision of **Sayefa Arafah** , Lecturer, Department of Computer Science and Engineering.

This project report has been accepted as satisfactory for the partial fulfillment of the requirement for the degree of **B.Sc. in Computer Science and Engineering** and approved as to its style and contents.

………………………

Supervisor
Sayefa Arafah
Lecturer
Department of Computer Science and Engineering
Bangladesh University of Business and Technology (BUBT)
Mirpur-2, Dhaka-1216, Bangladesh

………………………

Chairman
Md. Saifur Rahman
Associate Professor & Chairman
Department of Computer Science and Engineering
Bangladesh University of Business and Technology (BUBT)
Mirpur-2, Dhaka-1216, Bangladesh

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1. Problem Specification

In today's fast-paced digital world, children are often exposed to technology at a very early age. However, most learning tools are either too complex or not engaging enough for young minds. Traditional learning methods often fail to keep children interested, especially when it comes to learning about their everyday environment. This motivated us to develop **Tello**, an AI-powered smart learning toy that combines visual and audio-based learning in an interactive and fun way. The project was inspired by observing how children struggle to associate real-world objects with names, especially when learning new languages or concepts. Tello aims to bridge this gap using object detection, speech synthesis, and an intuitive display interface all within a compact IoT-enabled toy.

1.2. Project Objectives

- To develop a smart learning toy capable of recognizing real-world objects using AI and ESP32-CAM.
- To provide real-time speech output of detected object names using cloud-based text-to-speech technology.
- To display detected object names visually on an OLED screen.

1.3. Project Scope

- Integration of ESP32-CAM for capturing and recognizing objects in real time.
- Use of a cloud-based AI model (e.g., Edge Impulse) for object classification.
- Use of Google TTS API and MAX98357A audio amplifier for speech generation.
- Displaying object names via a 0.96" OLED screen for visual reinforcement.
- Building a Wi-Fi enabled system to support internet-based communication and updates.
- Designing the system to be power-efficient and portable using a rechargeable battery setup.
- Creating a modular and extensible system for future educational features.

1.4 Project Limitations

Tello has a few limitations. It recognizes only a limited set of voice commands, which may restrict user interaction. The hardware used is basic, limiting processing power and camera quality, especially in low-light conditions. Some features rely on a stable internet connection, and the system cannot understand emotions or context. Additionally, continuous use drains the battery quickly, limiting usage time.

CHAPTER 2

DESCRIPTION OF THE PROJECT

This project is focused on "**Tello** – AI-Powered Smart Learning Toy with IoT Connectivity" . Tello is built to engage children in an educational and playful way through audio-visual feedback, object recognition, and interaction. Unlike many existing screen-based systems, Tello aims to reduce screen time while encouraging physical interaction and learning.

The system uses real-time image recognition to identify objects, respond with context-aware audio messages using cloud-based Text-to-Speech (TTS), and interact with users through simple commands. It is capable of storing and retrieving interaction history using file handling, offering basic personalization.Tello has a friendly interface and supports both manual and automated modes of interaction. It can distinguish between different objects (like books, fruits, or toys), play learning sounds, and engage users in conversation-like behavior.

This project demonstrates how software can enhance early education by making technology more human-friendly, adaptive, and developmentally supportive for children. So,Tello offers,

1. Object Detection Module

- Uses the ESP32-CAM to capture images of surrounding objects.
- Sends images to a cloud-based AI model (Edge Impulse) for real-time recognition.
- Receives object names from the cloud for further processing.

2. Text-to-Speech (TTS) Module

- Converts detected object names into spoken words.
- Utilizes Google TTS API for natural and clear speech synthesis.
- Outputs audio through the MAX98357A amplifier and speaker.

3. Visual Feedback Module

- Displays detected object names on a 0.96” OLED screen.
- Provides visual confirmation to support audio output and reinforce learning.

CHAPTER 3
TECHNOLOGY & TOOLS

3.1 Equipments

Hardware:

- ESP32-CAM – Image capture and Wi-Fi communication
- MAX98357A – I2S Audio amplifier for TTS output
- OLED Display (0.96") – Visual feedback
- 2W 4Ω Speaker – Audio output
- Jumper Wires, Breadboard

Software:

- Arduino IDE – Programming microcontroller
- Edge Impulse
- Google TTS API – Text-to-speech conversion
- Cloud services – For AI processing and speech generation

CHAPTER 4
IMPLEMENTATION

DIAGRAM (Workflow):

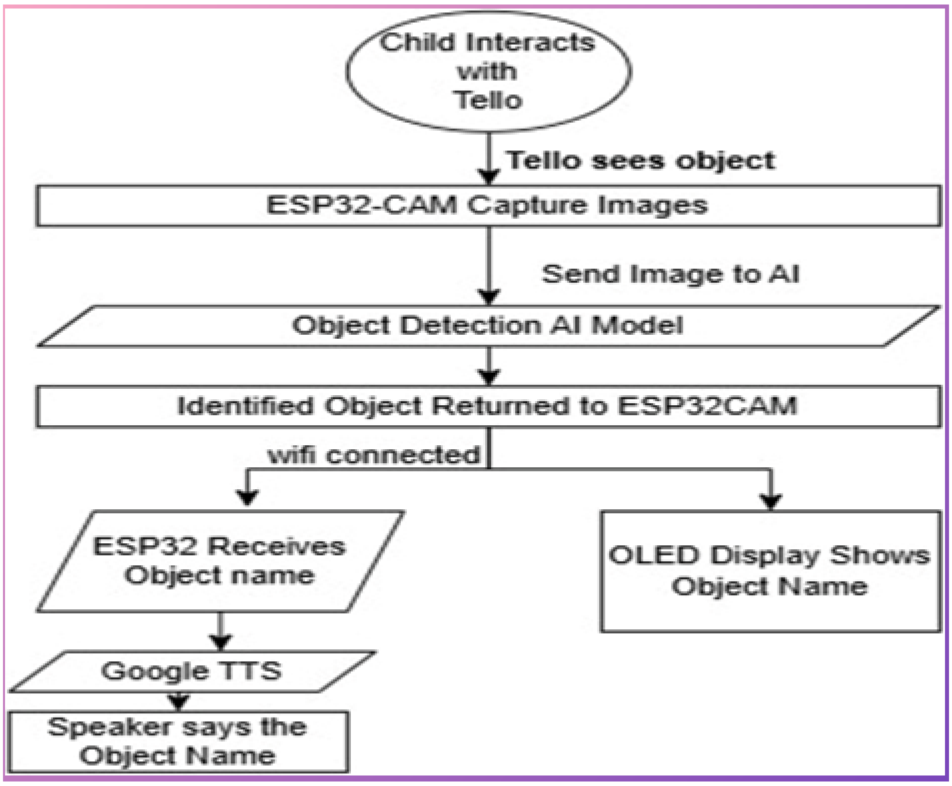


Fig:01

CONNECTIONS:

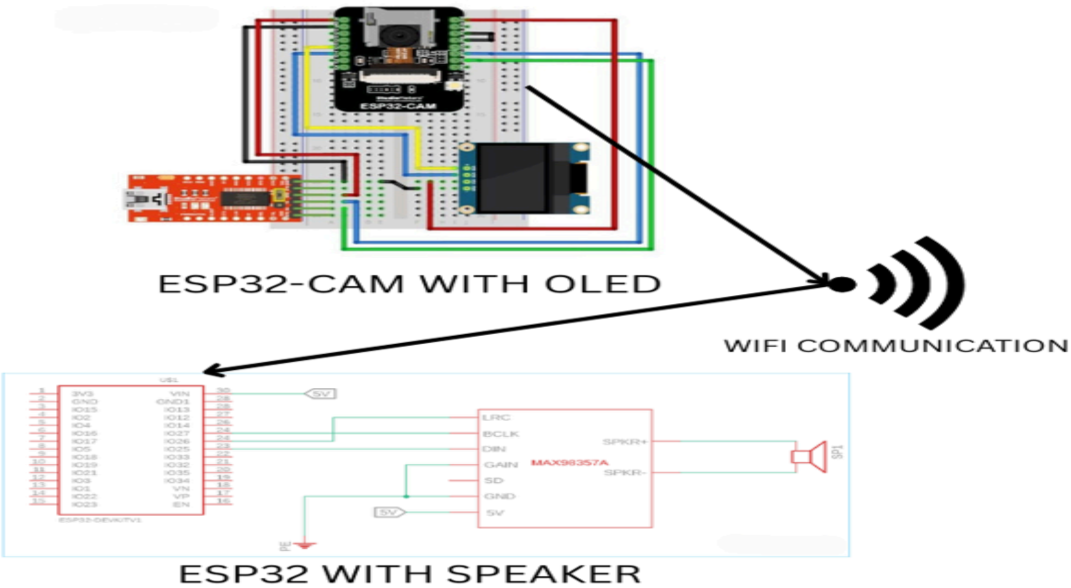


Fig:02

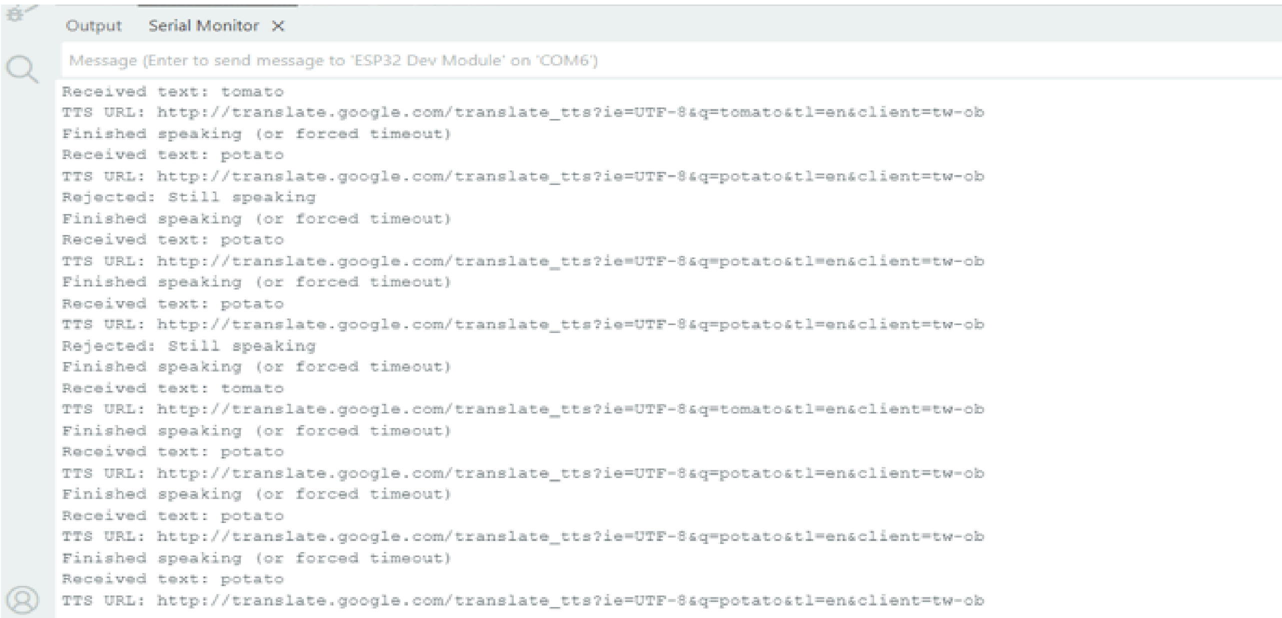
PROGRAM CODE

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <WebServer.h>
#include <Audio.h>

// WiFi credentials
const char* ssid = "TELLO";
const char* password = "12345678";

// I2S pin configuration
#define I2S_DOUT 25
#define I2S_BCLK 27
#define I2S_LRC  26

Audio audio;
WebServer server(80);
String currentTextToSpeak = "";
bool isSpeaking = false;
unsigned long speechStartTime = 0;
const unsigned long MAX_SPEAK_DURATION = 4000; // 4 seconds max per
message

void handleSpokenText() {
  if (server.hasArg("text")) {
    if (!isSpeaking) {
      currentTextToSpeak = server.arg("text");
      Serial.print("Received text: ");
      Serial.println(currentTextToSpeak);
      server.send(200, "text/plain", "Accepted: Speaking once");
    } else {
      server.send(429, "text/plain", "Busy: Already speaking");
      Serial.println("Rejected: Still speaking");
    }
  } else {
    server.send(400, "text/plain", "Error: 'text' argument not found");
  }
```

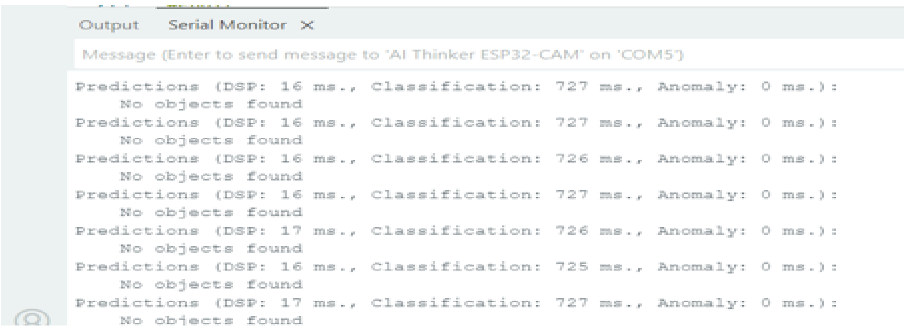
```
}

void setup() {
  Serial.begin(115200);
  WiFi.begin(ssid, password);

  Serial.println(F("Connecting to WiFi..."));
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("\nConnected to WiFi");
  Serial.print("ESP32 Speaker IP: ");
  Serial.println(WiFi.localIP());

  audio.setPinout(I2S_BCLK, I2S_LRC, I2S_DOUT);
  server.on("/speak", HTTP_GET, handleSpokenText);
  server.begin();
  Serial.println("HTTP server started");
}

void loop() {
  server.handleClient();
  audio.loop(); // Keep the audio running

  // Start speaking
  if (!isSpeaking && currentTextToSpeak != "") {
    playText(currentTextToSpeak);
    speechStartTime = millis();
    isSpeaking = true;
    currentTextToSpeak = "";  // Clear after sending
  }

  // Stop after natural finish OR timeout
  if (isSpeaking) {
        if (!audio.isRunning() || (millis() - speechStartTime >
MAX_SPEAK_DURATION)) {
      isSpeaking = false;
      Serial.println("Finished speaking (or forced timeout)");
```

```
    }
  }
}

void playText(const String& textToSpeak) {
  String chunk = textToSpeak;
  chunk.replace(" ", "%20");
  String tts_url = "http://translate.google.com/translate_tts?ie=UTF-8&q=" + chunk + "&tl=en&client=tw-ob";

  Serial.print("TTS URL: ");
  Serial.println(tts_url);

  audio.connecttohost(tts_url.c_str());
}
```

OUTPUT:



```
Output   Serial Monitor ×
Message (Enter to send message to 'ESP32 Dev Module' on 'COM6')
Received text: tomato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=tomato&tl=en&client=tw-ob
Finished speaking (or forced timeout)
Received text: potato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=potato&tl=en&client=tw-ob
Rejected: Still speaking
Finished speaking (or forced timeout)
Received text: potato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=potato&tl=en&client=tw-ob
Finished speaking (or forced timeout)
Received text: potato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=potato&tl=en&client=tw-ob
Rejected: Still speaking
Finished speaking (or forced timeout)
Received text: tomato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=tomato&tl=en&client=tw-ob
Finished speaking (or forced timeout)
Received text: potato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=potato&tl=en&client=tw-ob
Finished speaking (or forced timeout)
Received text: potato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=potato&tl=en&client=tw-ob
Finished speaking (or forced timeout)
Received text: potato
TTS URL: http://translate.google.com/translate_tts?ie=UTF-8&q=potato&tl=en&client=tw-ob
```



```
Output   Serial Monitor ×
Message (Enter to send message to 'AI Thinker ESP32-CAM' on 'COM5')
Predictions (DSP: 16 ms., Classification: 727 ms., Anomaly: 0 ms.):
    No objects found
Predictions (DSP: 16 ms., Classification: 727 ms., Anomaly: 0 ms.):
    No objects found
Predictions (DSP: 16 ms., Classification: 726 ms., Anomaly: 0 ms.):
    No objects found
Predictions (DSP: 16 ms., Classification: 727 ms., Anomaly: 0 ms.):
    No objects found
Predictions (DSP: 17 ms., Classification: 726 ms., Anomaly: 0 ms.):
    No objects found
Predictions (DSP: 16 ms., Classification: 725 ms., Anomaly: 0 ms.):
    No objects found
Predictions (DSP: 17 ms., Classification: 727 ms., Anomaly: 0 ms.):
    No objects found
```

PROJECT VIEW:

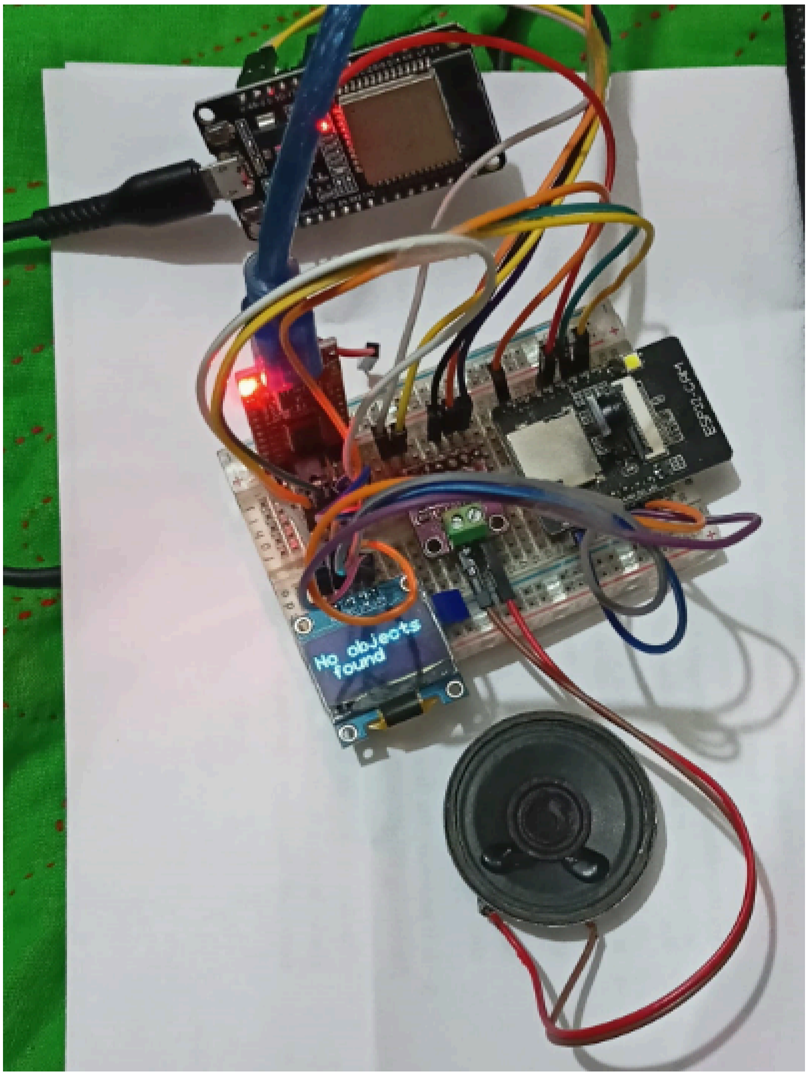


Fig:03

CHAPTER 5

CONCLUSION

5.1. Future Works

- Implement offline AI models to reduce dependency on cloud connectivity.
- Expand the object recognition database to cover more items.
- Add interactive learning features such as quizzes and progress tracking.
- Integrate parental controls and customization options.
- Enhance power management for longer battery life.

5.2 Conclusion

The “**Tello** – AI-Powered Smart Learning Toy with IoT Connectivity”,a smart learning toy successfully combines AI-powered object recognition with cloud-based text-to-speech (TTS) and interactive visual feedback to provide a modern, engaging, and educational experience for children. It addresses the limitations of traditional toys by introducing real-time intelligence, enhanced interactivity, and connectivity. This project demonstrated the effective integration of hardware components with cloud services, showcasing how AI and IoT can work together to support personalized learning. Through its development, we gained valuable insights into real-time AI processing, IoT connectivity, cloud-based communication, and embedded system programming, all of which contribute to creating smarter educational tools for the future.

References

1. Espressif Systems. (2021). ESP32 Technical Reference Manual. Espressif Systems.

2. Edge Impulse. (2023). Edge Impulse: Embedded Machine Learning Platform. Retrieved from https://www.edgeimpulse.com

3. Google Cloud. (2023). Text-to-Speech API Documentation. Retrieved from https://cloud.google.com/text-to-speech/docs

4. Maxim Integrated. (2018). MAX98357A I2S Audio DAC Datasheet. Maxim Integrated.

5. OLED Display Datasheet (0.96-inch). (2020). SSD1306 OLED Display Datasheet.

6. Arduino. (2023). Arduino IDE User Guide. Retrieved from https://www.arduino.cc/en/Guide

7. Szeliski, R. (2010). Computer Vision: Algorithms and Applications. Springer.

8. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

9. Hossain, M. S., Muhammad, G. (2019). Cloud-Assisted Industrial Internet of Things (IIoT)–Enabled Framework for Health Monitoring. IEEE Internet of Things Journal.

10. Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. IEEE Internet of Things Journal.