**A black background with blue text

Description automatically generated**

IOT-Barcode checker

Project report

**Students:**

Muhammad Biadsy

Omar Sharafy

**Supervised By:**

Mony Orbach

**Spring** 2024

Contents

[**1.** **Overview** 3](#_Toc172814372)

[**1.1** **Objective** 3](#_Toc172814373)

1. **Overview**

This project is a collaboration with Bnai Zion Medical Center aimed at addressing a common issue in hospital and medical center laboratories: the mismanagement of patient samples.

Currently, when samples are collected from patients, they are labeled with barcode stickers containing relevant patient information, such as name and ID. Upon arrival at the lab, these samples are checked against the patient's details in the computer system. Typically, staff label each sample with the patient’s name/ID and a unique barcode specific to the patient.

However, problems arise when multiple samples from the same or different patients arrive simultaneously. Manual handling can lead to confusion and mix-ups, making it crucial to verify at every stage that the samples belong to the correct patient and are intended for the designated laboratory.

In this project, we aim to design a low-cost, reliable barcode checker device that ensures all samples labeled as belonging to a patient (via name/ID) are correctly labeled with the matching barcode unique to the patient (a.k.a. "Golden Barcode").

# **2.inroduction**

## **2.1 Device Objective**

The goal of this project is to develop a compact and portable device that ensures all samples labeled with a patient’s name/ID have barcodes matching the patient's unique "Golden barcode."

This device will feature a 2D barcode scanner capable of reading the barcodes on the samples, providing immediate feedback on its screen and via sound to indicate whether the barcode on the sample matches the patient’s Golden barcode.

Additionally, the device will connect to a computer via Wi-Fi to display a list of the scanning results, ensuring comprehensive and accessible sample verification.

Designed to be user-friendly, cost-effective, and compatible with existing lab equipment and software, the device will be small size, chargeable via USB-Type C & Showing results Via sound and Screen to avoid any confusion.

## **2.2 Device Components**

For the objectives this project aims to achieve, we used a list of components to help with our design.

### **2.2.1 ESP-32**

ESP32 is a series of low cost, low-power system on a chip microcontroller with integrated Wi-Fi and dual-mode Bluetooth.

Cost: 1$

### **2.2.2 Display ILI9341**

3.2-inch display that is used to display

important information about the barcode matching.

Resolution: 320X240 pixels

Comm. Protocol: SPI

Cost: 6.5$

### A black square with blue and white lights Description automatically generated**2.2.3 Barcode scanner Grow Gm810**

5.6 cm wide barcode scanner uses UART protocol.

Scans the barcode and provide us with important information about it.

Comm. Protocol: UART

Cost: 18$

### **2.2.4 RTC**

A real time clock with high accurce, will be use to recored the time of each scanning and comparing operation and save it in order to allow backtrace.

Cost: 1$

### **2.2.5 SD Card**

A high-accuracy real-time clock will be used to record the time of each scanning and comparison operation, ensuring precise timestamping. This data will be saved to facilitate backtracking and auditing of all activities.

Comm. Protocol: H-SPI.

Cost: 1$

The overall cost of the components and sensor for the device sums up to about 27.5$ US Dollars, in addition to other components we will talk about and manufacturing cost, we can sum up to about 45$ US Dollars which is a noticeably a low price in comparison to other devices in the market.

## **2.3 Development Environments**

### **2.3.1 Arduino IDE**

Used to write and upload the code to ESP-32 control board.

### A close up of a logo Description automatically generated **2.3.1 OrCad**

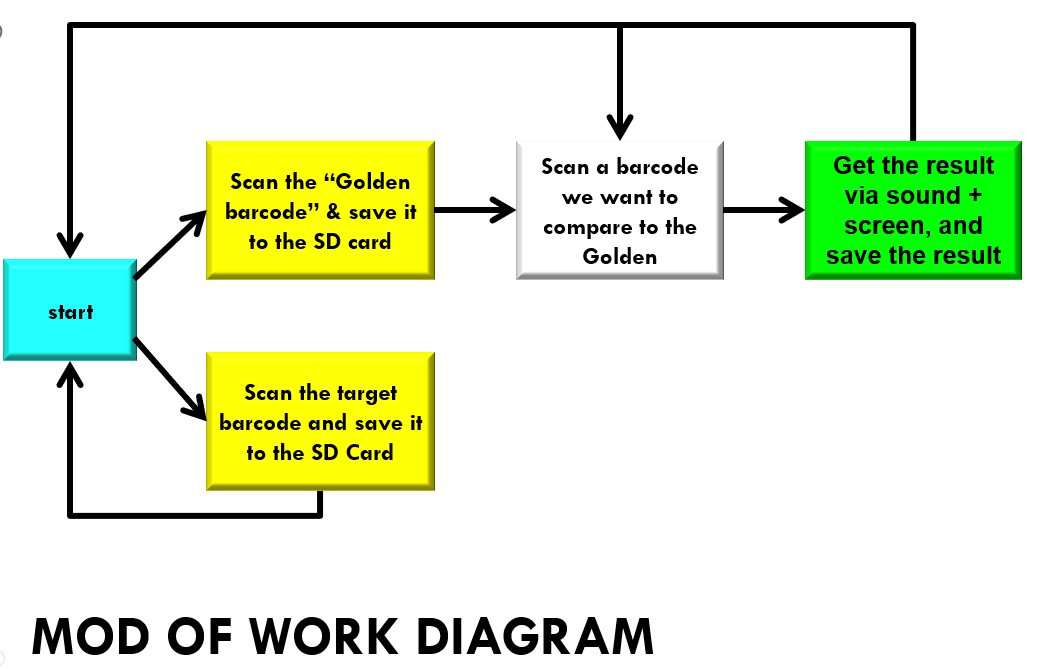
Used for designing the electric schematic and it’s blocks.

### **2.3.1 GrebV**

Used for viewing the Greber files.

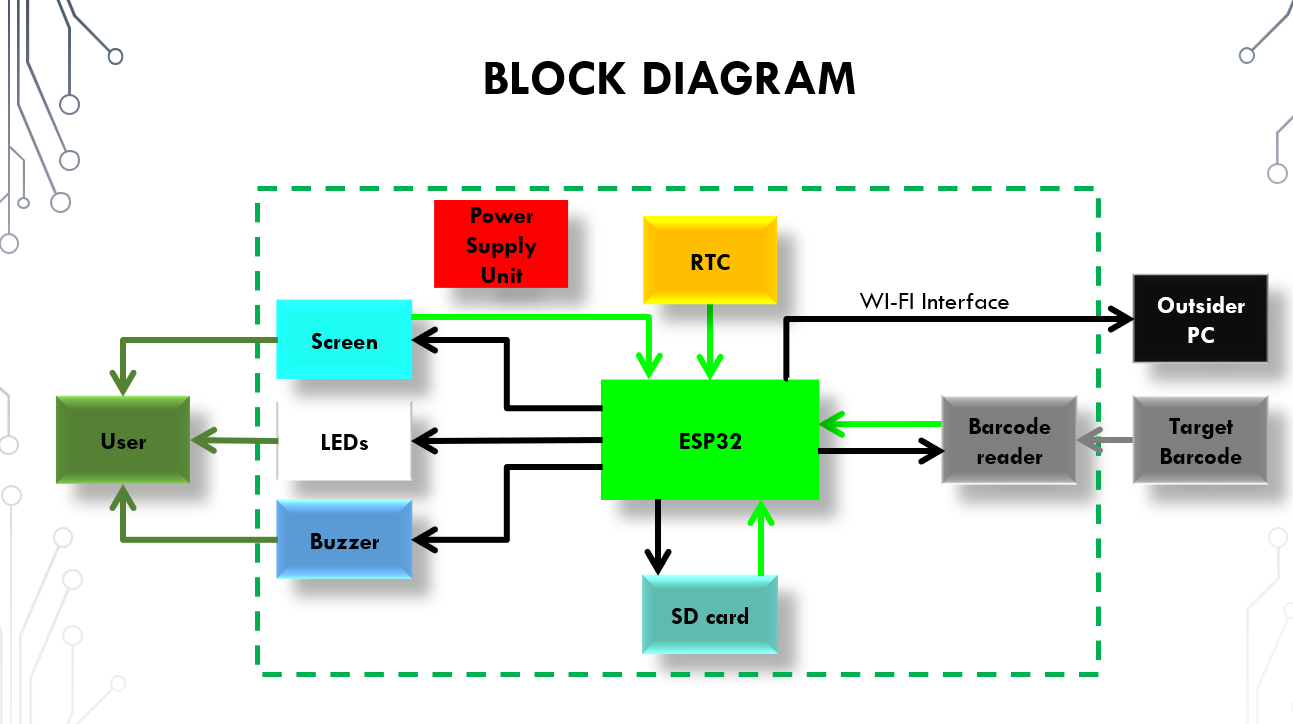
# **3.Flow Diagram**

As described before, the device functionality starts by Clicking a button indicating Triggering the wanted Work mode, then the user Scans the target Barcode Using the built in scanner. //should talk to mony.



# **4.Block Diagrams**

A top-view block diagram of the Bar-code checker components and connection both internally between the sensors and microprocessor, and externally with the outside world.



## **4.1 Controller Unit**

## **4.2 Power Management Unit**

This unit is responsible for supplying the voltages for the different components of the design.

Note that the Decision on whenever a certain component is working and gets its required voltage is managed in the Control Unit.

A diagram of power management

Description automatically generated

## **4.3 Sensors**

As explained at the introduction, a list of sensors are being used to help with the functionality of the station. These sensors are connected to the micro processing unit (ESP32) via various communication protocols.

## **4.4 Communication Map**