QR-code and Bar-code Scanner

Aim of the project:

Solve a typical problem of Barcode +QR Code Reader using Arduino & QR Scanner Module.

PROBLEM STATEMENT AND SOLUTION:

There is a need for an efficient barcode and QR code reader system that is both reliable and cost-effective. However, traditional QR scanner modules are expensive and not easily available.

To address this challenge, we propose a solution utilizing Arduino and the ESP32-CAM module. This alternative offers affordability and availability without compromising on accuracy. By leveraging Arduino's flexibility and the ESP32-CAM's capabilities, our system ensures reliable decoding and processing of barcode and QR code data.

PROJECT DESIGN SPECIFICATIONS:

- **Compatibility**: Ensure compatibility with various types of barcodes (1D) and QR codes (2D) commonly used in the market, such as UPC, EAN, Code 39, Code 128, etc., as well as standard QR code formats.

- **Accuracy**: Achieve high accuracy in code recognition to minimize errors in decoding and ensure reliable data capture.

- **Scanning Speed**: Aim for efficient scanning performance with minimal delay between code detection and processing, ensuring a smooth user experience.

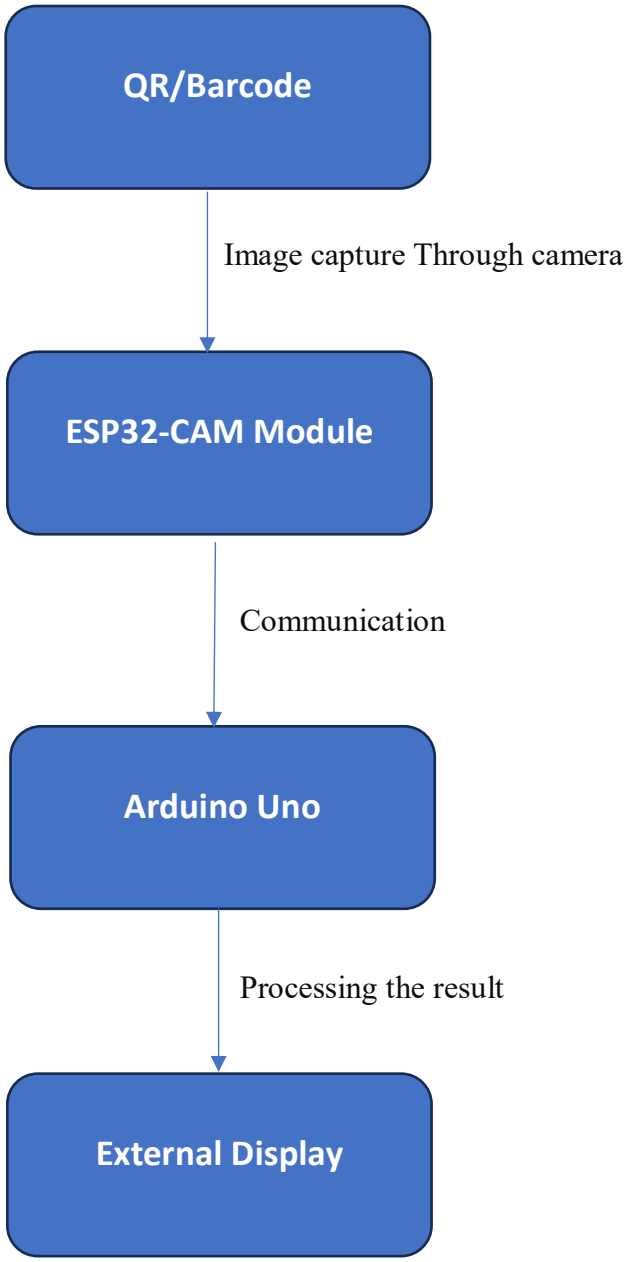
- **Resolution**: Support scanning of codes with varying resolutions to accommodate codes printed in different sizes and qualities.

- **Durability**: Consider the durability and robustness of the hardware components and enclosure, especially if the project is intended for industrial or outdoor use.

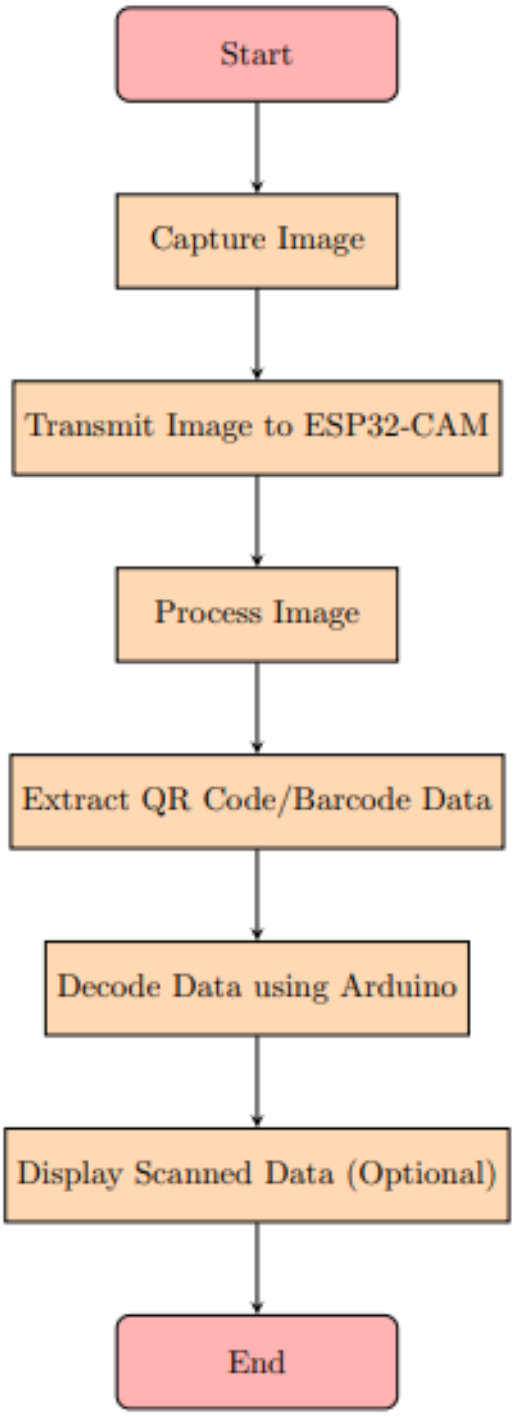
- **Cost**: Aim for a balance between performance and cost-effectiveness, selecting components and materials that meet the project requirements without unnecessary expense.

By abiding to these design specifications, you can develop a Barcode + QR code reader system that effectively solves the typical problems associated with such technology using Arduino and a QR scanner module.

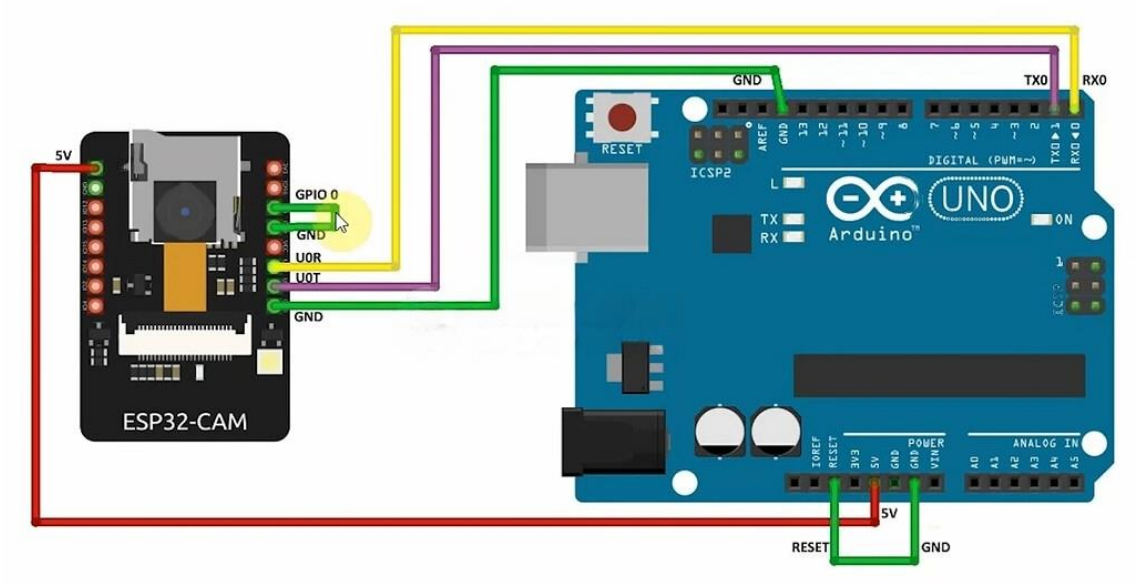
PROJECT ARCHITECTURE:



FLOW EXPLANATION:



Wiring Daigram:-



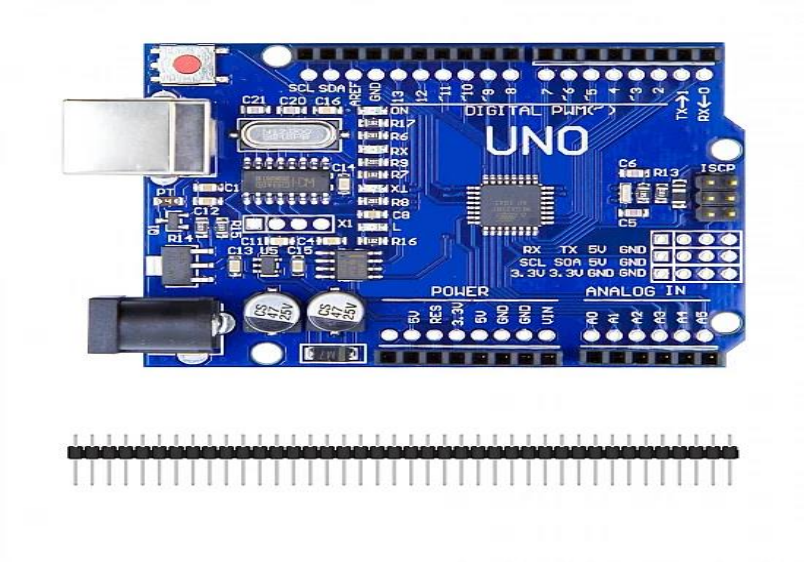
COMPONENTS WORKING PRINCIPLES / FUNCTIONALITY:

ESP32-CAM:

The ESP32-CAM integrates the ESP32 microcontroller and camera module, offering Wi-Fi and Bluetooth connectivity alongside image capture capabilities. With dual-core processors, GPIO pins, and camera modules like OV2640 or OV7670, it enables real-time image capture and transmission for applications such as surveillance and visual data logging.

Arduino Uno:

The Arduino Uno, based on the ATmega328P microcontroller, serves as a versatile open-source platform for interfacing with electronic components. Its simple programming environment allows users to control sensors and actuators via digital and analog pins. Operating at 5 volts, the Uno communicates with peripherals, making it ideal for a wide range of projects from basic prototyping to advanced automation.



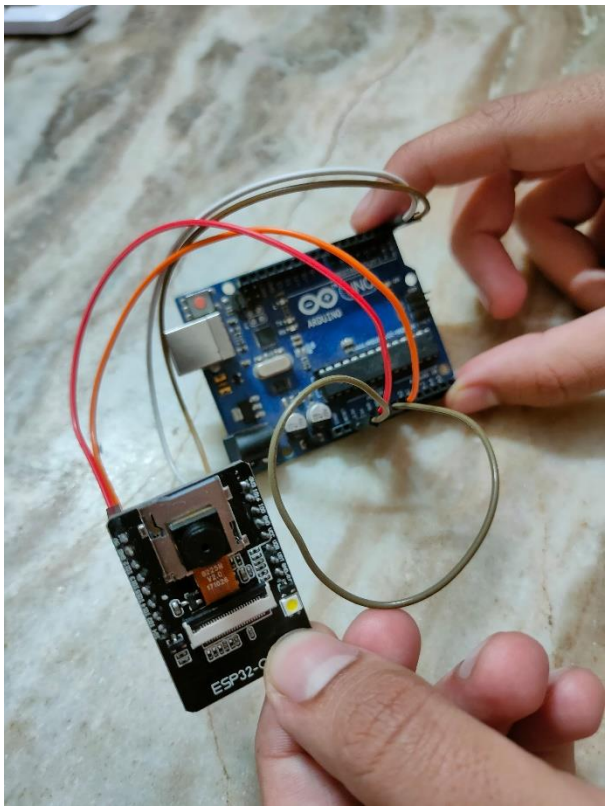
Jumper Wire:

Jumper wires are flexible electrical wires with connectors at each end, facilitating electrical connections between electronic components. They are crucial for quick prototyping and experimentation, allowing for temporary connections without soldering. Available in various lengths and colors, jumper wires streamline circuit construction and enable rapid iteration in electronics projects.

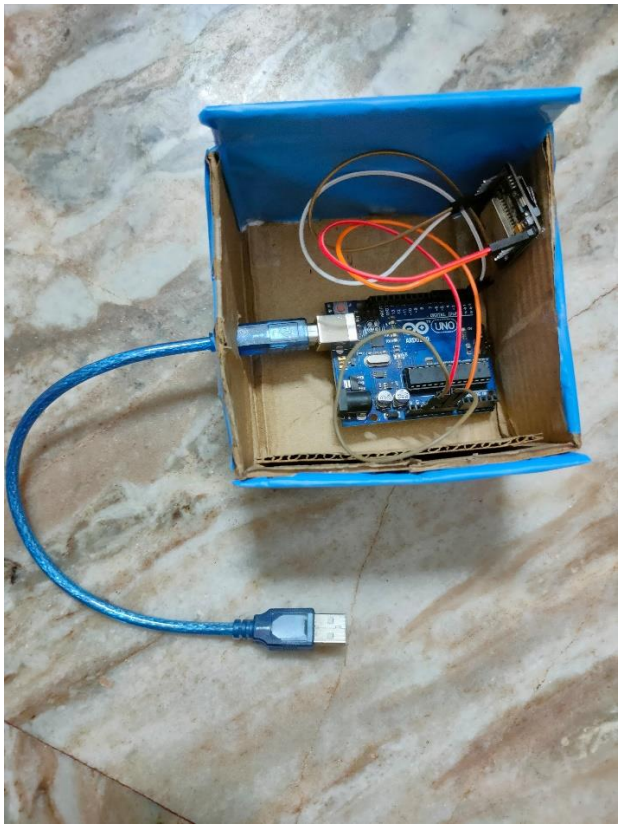
Arduino IDE:

The Arduino Integrated Development Environment (IDE) is a user-friendly software tool for writing, compiling, and uploading code to Arduino boards. It simplifies the process of programming microcontrollers like the Arduino Uno by providing an intuitive interface for writing and editing code. With built-in libraries and examples, the IDE offers extensive support for beginners and experts alike, making it an essential tool for developing projects ranging from simple blinking LEDs to complex IoT applications.

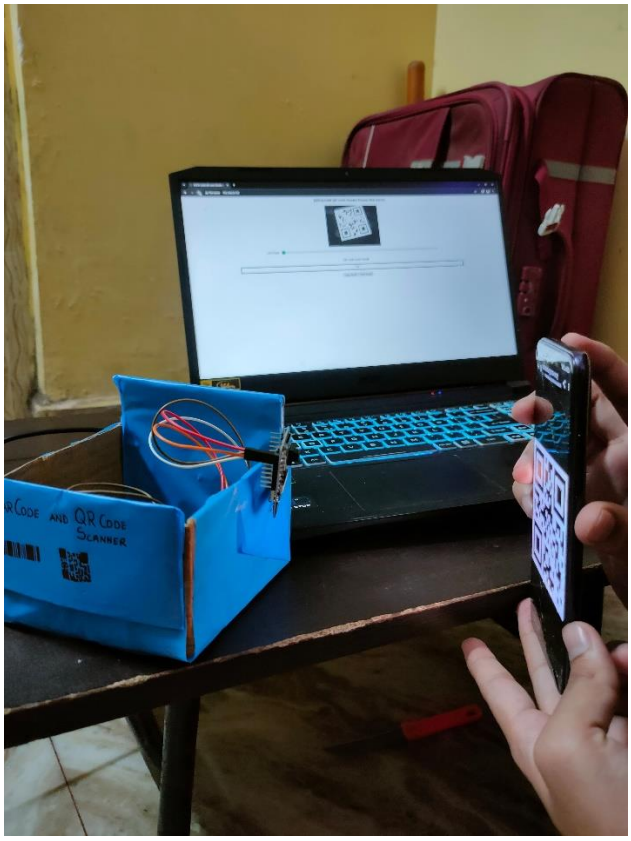
Hardware implementation:

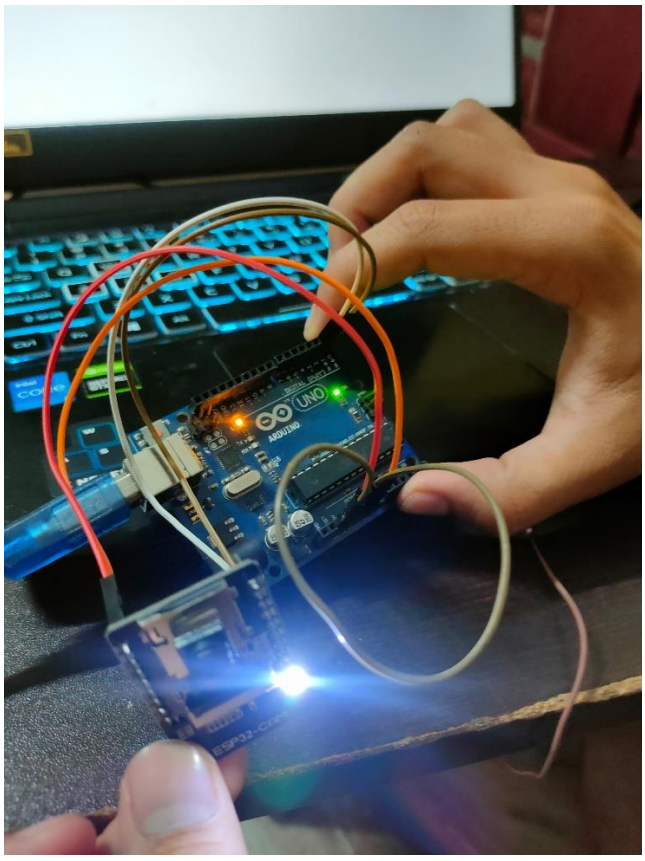


Hardware model:



Working model:





Output:


Barcode + QR Code Reader Stream Web Server
LED Flash
Scan Result :
610762569327
Copy Result
Clear Result


Barcode + Code Reader Stream Web Server
LED Flash
Scan Result :
Hello world!
Copy Result
Clear Result


Barcode + QR Code Reader Stream Web Server
LED Flash
Scan Result :
http://en.m.wikipedia.org
Copy Result
Clear Result

Bill of the material preperration:

SN	Device/Instrument	Model	Qunatity	EST unit Price	Total
1	ESP32-cam	ESP32-S	1	₹450	₹450
2	Arduino	UNO	1	₹520	₹520
3	Jumper wire	Male-female	5	₹10	₹10
4	miscellaneous	-	-	₹30	₹30
TOTAL					₹1010

Project Outcome:

Understanding the problem statements in the STP:

At the outset of our project, we delve into a comprehensive understanding of the challenges specified in the System Test Plan (STP). This involves carefully identifying the needs and barriers associated with the development of a QR code and barcode scanner. Specifically, our focus lies on the utilization of a QR scanner module in conjunction with Arduino Uno. Key considerations include ensuring efficient image processing, precise decoding capabilities for barcodes, and seamless integration of the QR scanner module with Arduino Uno.

Finding out the best solutions:

After understanding the problem statements, the project shifts its focus to identifying and evaluating the most suitable solutions for implementing a QR code and barcode scanner using a QR scanner module and Arduino Uno. Recognizing the challenges posed by the expensive and limited availability of QR scanner modules, the project proactively explores alternatives. Through thorough research into available libraries and modules compatible with both the QR scanner module and Arduino Uno, the project identifies the ESP32-CAM as a cost-effective and readily available alternative. This assessment includes considerations of features, performance, and ease of integration to ensure the chosen solution aligns seamlessly with the project's requirements.

Wire connection and verification:

After selecting the ESP32-CAM as the optimal solution, the project proceeds to physically connect it to the Arduino Uno. This involves carefully attaching jumper wires between specific pins on the ESP32-CAM and corresponding pins on the Arduino Uno board. For instance, the TX (transmit) pin of the ESP32-CAM might be connected to the RX (receive) pin of the Arduino Uno, and vice versa, to establish serial communication between the two devices. Additionally, GPIO (General Purpose Input/Output) pins on both the ESP32-CAM

and Arduino Uno may be interconnected to enable communication and control between the modules.

Furthermore, power and ground connections are established to ensure proper functioning of the components. The ESP32-CAM typically requires a 5V power supply, which can be sourced from the Arduino Uno's 5V pin. However, it's essential to verify the power requirements of the ESP32-CAM and ensure compatibility with the Arduino Uno's power output.

Once all connections are made, thorough verification and testing procedures are conducted. This involves visually inspecting the connections to ensure they are secure and correctly aligned. Additionally, functional tests are performed to verify that data can be transmitted between the ESP32-CAM and Arduino Uno through the established communication interfaces. This includes sending test signals or commands from the Arduino Uno to the ESP32-CAM and verifying that the expected responses are received.

Components Exploration:

As part of the project, we extensively explore the components involved, including the ESP32-CAM and Arduino Uno. This involves understanding their specifications, functionalities, and limitations. We also research QR code and barcode scanning techniques suitable for the ESP32-CAM, focusing on efficient decoding methods. Additionally, we experiment with various image processing libraries and algorithms to optimize the decoding process for accurate QR code and barcode recognition. This exploration phase allows us to make informed decisions and develop effective solutions for QR code and barcode scanning using the ESP32-CAM and Arduino Uno.

Purpose of Integration of sensors and actuators with microcontroller unit:

The final aspect of the project outcome revolves around the purposeful integration of ESP32-CAM with Arduino Uno. By seamlessly integrating ESP32-CAM with Arduino Uno, the scanner system gains the capability to capture images, process them for QR codes and barcodes, and initiate appropriate actions based on the scanned information. This integration enables the development of a robust scanning system capable of interfacing with external devices, executing complex image processing tasks, and facilitating real-time data analysis.