System Design Document – Smart Secure Access Gate

1. Introduction

The Smart Secure Access Gate is designed to control entry to authorized personnel while ensuring safety and recording all access events. The system combines **real-time control via Arduino** with **high-level processing, authentication, and monitoring via Raspberry Pi**. Key functionalities include user authentication, door control, safety monitoring, real-time alerting, and remote access to logs and live video. This document evaluates multiple design options considered for the project, describing each option, evaluating it across relevant metrics, and justifying the final choice.

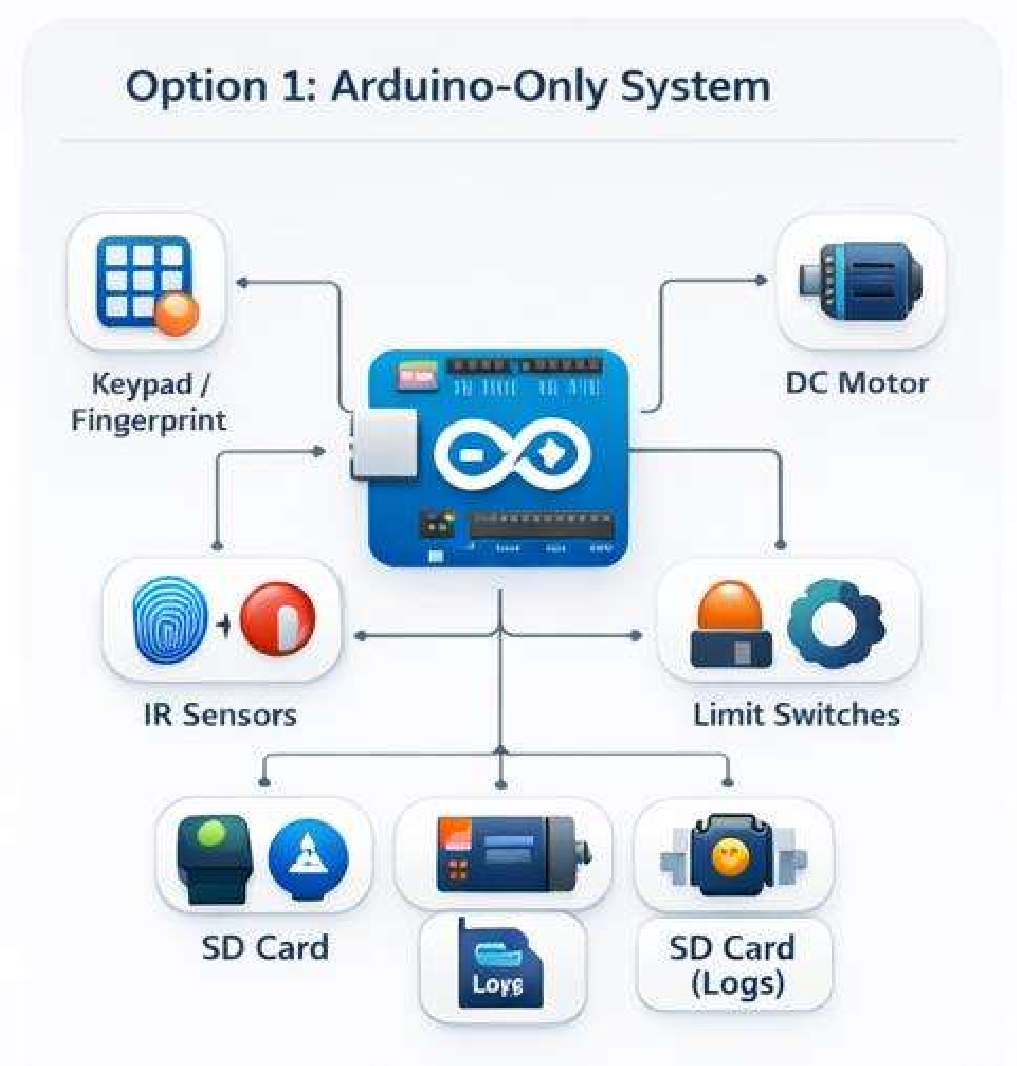
2. Design Options

Option 1: Arduino-Only System

Description:

- The entire system, including user authentication, door control, safety monitoring, and event logging, is implemented solely on an Arduino Uno.

- **Components:**
  - Arduino Uno
  - Keypad / Fingerprint sensor for authentication
  - DC Motor with motor driver
  - IR beam sensors, limit switches, current sensor, emergency stop
  - Buzzer / LED for alerts
  - Optional SD card for local storage of access logs
- **Block Diagram :**

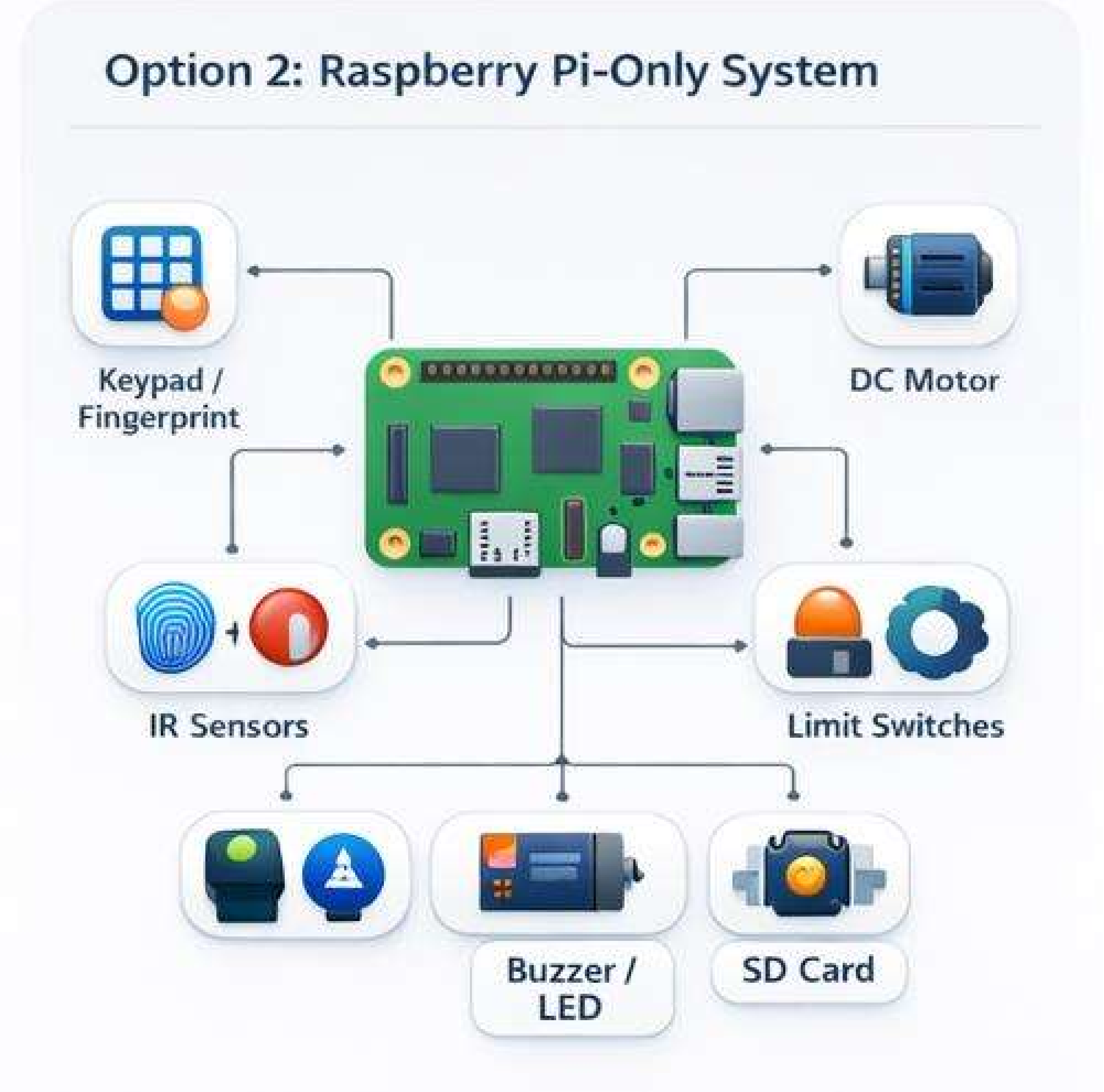


Option 2: Raspberry Pi-Only System

- **Description:**

The Raspberry Pi handles all functionalities, including authentication, motor control, safety monitoring, and video capture.

- **Components:**
  - Raspberry Pi 4
  - Keypad / Fingerprint sensor
  - Motor driver + DC motor
  - IR beam sensors, limit switches, current sensor
  - IP Camera
  - Buzzer / LED
- **Block Diagram:**



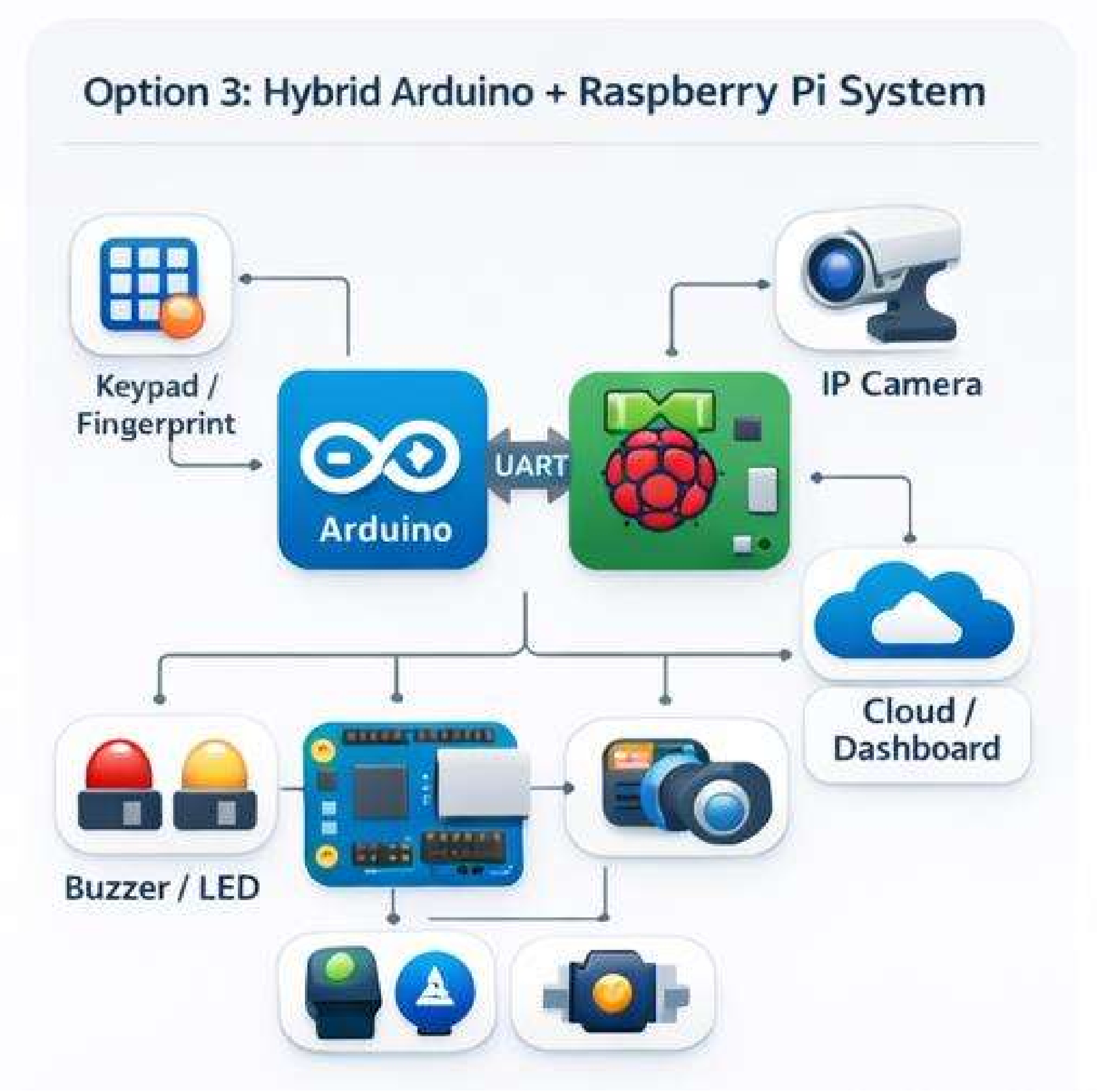
Option 3: Hybrid Arduino + Raspberry Pi System (Chosen Option)

- **Description:**

Combines the strengths of Arduino for **real-time motor control and safety monitoring** with Raspberry Pi for **authentication, video processing, event logging, and remote access**.

- **Components:**
  - Arduino Uno (real-time control)
  - Raspberry Pi 4 (authentication, logging, video)
  - Keypad / Fingerprint sensor
  - DC Motor with driver
  - IR beam sensors, limit switches, current sensor, emergency stop
  - IP Camera
  - Buzzer / LED

- **Block Diagram:**



3. Evaluation of Options

Metric	Option 1: Arduino-Only	Option 2: Raspberry Pi-Only	Option 3: Hybrid (Chosen)
Real-Time Response	Excellent	Poor	Excellent (Arduino handles)
Video Processing	Not feasible	Feasible	Feasible (Pi handles)
Safety Monitoring	Excellent	Moderate (Pi may delay response)	Excellent (Arduino handles)
Complexity	Low	High	Moderate
Reliability	High (limited tasks)	Moderate	High
Remote Access / IoT	Limited (SD card)	Full	Full (Pi handles)
Ease of Implementation	Easy	Hard	Moderate

Evaluation Notes:

1. **Option 1** is simple and highly reliable for real-time control but cannot handle video processing or remote monitoring effectively.
2. **Option 2** centralizes all control on Raspberry Pi, but real-time safety is less reliable due to OS latency, and motor safety risks increase.
3. **Option 3** provides the best of both worlds: Arduino ensures **real-time motor control and safety**, while Raspberry Pi handles **authentication, video capture, logging, and IoT functionality**.

4. Choice Justification

The **Hybrid Arduino + Raspberry Pi system** was selected because it maximizes both **safety and functionality**:

- Real-time motor control and safety monitoring remain reliable via Arduino.
- Raspberry Pi handles authentication, logging, and live/recorded video streaming, providing **full remote access**.
- The hybrid design allows **scalability and maintainability** for future enhancements (e.g., AI-based intrusion detection).
- Complexity is manageable, and the system meets all project requirements, including **real-time safety, IoT integration, and multi-user monitoring**.

5. Conclusion

The selected hybrid system is a **robust, secure, and scalable solution** for a smart access gate. It provides **real-time safety**, **user authentication**, **video monitoring**, and **remote accessibility**, satisfying both **functional and safety requirements** of the Capstone project.

Document Information

This System Design Document was prepared for the course
Programming for the Internet of Things – Capstone Project,
University of California, Irvine (UCI).

Instructor: Ian Harris   Student: Hamed Javadi Dafsari
Date: 2025/12/31