|  |  |
| --- | --- |
|  | **TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  **\*\*\*\*\*\***  **KHOA ĐIỆN - ĐIỆN TỬ**  BỘ MÔN KỸ THUẬT ĐIỆN TỬ <http://dee.utc.edu.vn/> |



**BÁO CÁO MÔN HỌC ITS**

**Chủ đề:**

**Mô hình bãi đỗ xe ứng dụng công nghệ nhận dạng RFID**

**Họ Và Tên: Lê Văn Đức \_ 181403463**

**Lương Đức Nguyên \_ 181401866**

**Lớp: Kĩ Thuật Điện Tử và Tin Học Công nghiệp 1 – K59**

**GV Hướng Dẫn: Ts. Trần Văn Hưng**

MỤC LỤC

[Chương 1: TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc105175717)

[1.1.TỔNG QUAN 1](#_Toc105175718)

[1.2. ĐẶT VẤN ĐỂ 1](#_Toc105175719)

[1.3. MỤC ĐÍCH LUẬN ÁN 2](#_Toc105175720)

[1.4. KẾT LUẬN CHƯƠNG I 2](#_Toc105175721)

[1.4.1.Nội dung nghiên cứu 2](#_Toc105175722)

[1.4.2. Giới hạn 3](#_Toc105175723)

[1.4.3. Phạm vi sử dụng 3](#_Toc105175724)

[Chương 2: THIẾT KẾ VÀ THỰC HIỆN CHƯƠNG TRÌNH 4](#_Toc105175725)

[2.1. THIẾT KẾ TỔNG QUÁT 4](#_Toc105175726)

[2.2.1. Yêu cầu của hệ thống 4](#_Toc105175727)

[2.2.2. Sơ đồ khối và chức năng mỗi khối 4](#_Toc105175728)

[2.2. THIẾT KẾ CHI TIẾT 5](#_Toc105175729)

[2.2.1. Phần cứng 5](#_Toc105175730)

[2.2.2. Lập trình 13](#_Toc105175734)

[2.3. TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG PHẦN CỨNG 16](#_Toc105175735)

[2.3.1. Khối chấp hành: 16](#_Toc105175736)

[2.3.2. Khối hiển thị: LCD\_I2C 17](#_Toc105175737)

[2.3.3. Khối Reader 18](#_Toc105175738)

[2.3.4. Khối xử lý trung tâm: 19](#_Toc105175739)

[2.3.5. Khối nguồn 20](#_Toc105175740)

[2.4. LƯU ĐỒ GIẢI THUẬT HỆ THỐNG 21](#_Toc105175741)

[2.4.1. Lưu đồ giải thuật chương trình chính 21](#_Toc105175742)

[2.4.2. Lưu đồ chương trình con RFID 22](#_Toc105175743)

[2.5. SẢN PHẨM HOÀN THIỆN 23](#_Toc105175744)

[2.5.1. Mô hình RFID 23](#_Toc105175745)

[2.5.2. Chương trình Arduino IDE 23](#_Toc105175746)

[2.5.3. Phần mềm Excel và công cụ PLX-DAQ 24](#_Toc105175747)

[2.6. KẾT LUẬN CHƯƠNG II 24](#_Toc105175748)

[Chương 3: THỬ NGHIỆM MẠCH THỰC VÀ KẾT QUẢ 25](#_Toc105175749)

[3.1. KỊCH BẢN THỬ NGHIỆM 25](#_Toc105175750)

[3.2. KẾT QUẢ THỰC NGHIỆM 26](#_Toc105175751)

[3.3. ĐÁNH GIÁ KẾT QUẢ 29](#_Toc105175752)

[Chương 4: KẾT LUẬN CHUNG VÀ HƯỚNG PHÁT TRIỂN 30](#_Toc105175753)

[4.1. KẾT LUẬN CHUNG 30](#_Toc105175754)

[4.2. HƯỚNG PHÁT TRIỂN 30](#_Toc105175755)

**DANH MỤC CÁC TỪ VIẾT TẮT**

LCD: Liquid-Crystal Display

I2C: Inter-Integrated Circuit

RFID: Radio Frequency Identification

ID: Identification

IDE: Integrated Development Environment

GPIO: General Purpose Input Output

UART: Universal Asynchronous Receiver Transmitter

SPI: Serial Peripheral Interface

PWM: Pulse Width Modulation

SCL: Serial Clock

SDA: Serial Data

RX: Receive

TX: Transmitter

IC: Integrated Circuit

USB: Universal Serial Bus

**DANH MỤC CÁC BẢNG BIỂU**

[Bảng 2. 1: Bảng thông số kĩ thuật Arduino Uno R3 6](#_Toc100264718)

[Bảng 2. 2: Bảng thông số kĩ thuật 7](#_Toc100264719)

[Bảng 2. 3: Thông số kĩ thuật LCD 16x2 9](#_Toc100264720)

[Bảng 2. 4: Thông số kĩ thuật Module I2C 11](#_Toc100264721)

[Bảng 2. 5: Thông số kĩ thuật buzzer 12](#_Toc100264722)

[Bảng 2. 6: Thông số kĩ thuật động cơ Servo 13](#_Toc100264723)

[*Bảng 2. 7: Sơ đồ kết nối chân MFRC522 với Arduino …………………………………….. 19*](#_Toc100264827)

[Bảng 2. 8: Dòng và áp của các linh kiện được sử dụng trong hệ thống 20](#_Toc100264828)

**DANH MỤC CÁC HÌNH ẢNH, BIỂU ĐỒ**

[Hình 2. 1: Hình ảnh thực tế Arduino UNO R3 SMD 5](#_Toc100334687)

[Hình 2. 2: Kí hiệu Ardunio UNO R3 SMD 5](#_Toc100334688)

[Hình 2. 3: Sơ đồ chân Arduino UNO R3 SMD 5](#_Toc100334689)

[Hình 2. 4: Hình thực tế Module RFID RC522………………………………………………….6   
Hình 2. 5: Kí hiệu MFRC522 7](#_Toc100334690)

[Hình 2. 6:Hình ảnh thực tế LCD 16x2 8](#_Toc100334691)

[Hình 2. 7: Kí hiệu LCD 16x2 8](#_Toc100334692)

[Hình 2. 8: Sơ đồ điều khiển LCD 16x2 9](#_Toc100334693)

[Hình 2. 9: Hình thực tế module I2C 11](#_Toc100334694)

[Hình 2. 10: Sơ đồ các chân I2C 11](#_Toc100334695)

[Hình 2. 11: Kí hiệu I2C 11](#_Toc100334696)

[Hình 2. 12: Hình thực tế buzzer 12](#_Toc100334697)

[Hình 2. 13: Sơ đồ các chân buzzer 12](#_Toc100334698)

[Hình 2. 14: Kí hiệu buzzer 12](#_Toc100334699)

[Hình 2. 15: Hình thực tế động cơ Servo SG90 12](#_Toc100334700)

[Hình 2. 16: Sơ đồ các chân động cơ Servo SG90 12](#_Toc100334701)

[Hình 2. 17: Kí hiệu động cơ Servo SG90 12](#_Toc100334702)

[Hình 2. 18:Giao diện PHẦN MỀM ARDUINO IDE 14](#_Toc100334703)

[Hình 2. 19: Giao diện kết nối giữa Arduino và Excel 15](#_Toc100334704)

[Hình 3. 1: Dữ liệu xe vào bến 25](#_Toc105176418)

[Hình 3. 2: Dữ liệu xe ra bến 26](#_Toc105176419)

[Hình 3. 3: Mô hình mạch testboard của đề tài 27](#_Toc105176420)

[Hình 3. 4: trạng thái LCD 27](#_Toc105176421)

[Hình 3. 5: Trạng thái LCD khi quét ID thẻ đúng ở ngõ vào 27](#_Toc105176422)

[Hình 3. 6: Trạng thái LCD khi quét ID thẻ sai ở ngõ vào 27](#_Toc105176423)

[Hình 3. 7: Trạng thái LCD khi quét ID thẻ đúng ở ngõ ra 28](#_Toc105176424)

[Hình 3. 8: Trạng thái LCD khi quét ID thẻ sai ở ngõ ra 28](#_Toc105176425)

[Hình 3. 9: Bảng thiết lập kết nối giữa bộ xử lí trung tâm và file Excel trên máy tính 28](#_Toc105176426)

[Hình 4. 1: Mô hình mạch testboard của đề tài 27](#_Toc100334713)

[Hình 4. 2: trạng thái LCD 27](#_Toc100334714)

[Hình 4. 3: Trạng thái LCD khi quét ID thẻ đúng ở ngõ vào 27](#_Toc100334715)

[Hình 4. 4: Trạng thái LCD khi quét ID thẻ sai ở ngõ vào 27](#_Toc100334716)

[Hình 4. 5 : Trạng thái LCD khi quét ID thẻ đúng ở ngõ ra 28](#_Toc100334717)

[Hình 4. 6: Trạng thái LCD khi quét ID thẻ sai ở ngõ ra 28](#_Toc100334718)

[Hình 4. 7: Bảng thiết lập kết nối giữa bộ xử lí trung tâm và file Excel trên máy tính 28](#_Toc100334719)

**Chương 1: TỔNG QUAN VỀ ĐỀ TÀI**

**1.1.TỔNG QUAN**

Hiện nước ta đang phát triển theo hướng công nghiệp hóa hiện đại hóa, nhu cầu của con người về sự thuận lợi trong cuộc sống ngày càng tăng, mật độ dân cư và xe cộ càng ngày càng đông. Đặc biệt là sự gia tăng về số lượng xe, điều này cũng phản ánh sự phát triển của một quốc gia nhưng cũng hướng dẫn những vấn đề như môi trường, tắc nghẽn giao thông, thiếu bãi đậu xe, cần được giải quyết thiết bị cấp.

Trong khi ngành công nghiệp điện tử cũng đang phát triển rất mạnh, đã và đang dần đáp ứng được nhu cầu từ tất cả các lĩnh vực công, nông, lâm, ngư nghiệp cho đến những nhu cầu trong đời sống hàng ngày. Với trình độ khoa học kỹ thuật ngày càng phát triển, nhiều vấn đề từ đó được giải quyết nhanh gọn với công nghệ điện tử và tự động hoá.

Việt Nam là một nước có hệ thống bảo mật, tự động trong các hệ thống quản lí còn thô sơ trong khi đó các nước phát triển đã tiến hành tự động hóa nhằm giảm lực lượng lao động, tăng hiệu quả công việc một cách tối đa. Điển hình là trong một hình thức giữ xe truyền thống, theo phương thức cũ người dân phải bỏ ra nhiều công sức, nhân công lao động để điều hành một hệ thống giữ xe, điều này dẫn đến việc bỏ ra nhiều chi phí cho nhân công và quản lí. Trong khi xã hội ngày càng tiến bộ, kỹ thuật ngày càng tiên tiến hơn, thì nhu cầu đòi hỏi về những thiết bị ứng dụng thông minh, tự động càng nhiều hơn. Và với mong muốn ứng dụng công nghệ RFID vào việc bảo mật và các cảm biến để thực hiện các tác vụ một cách tự động, tiết kiệm sức lao động và đạt được hiệu quả cao, em đã chọn đề tài: **“Bãi xe tự động sử dụng công nghệ RFID”** để thực hiện.

**1.2. ĐẶT VẤN ĐỂ**

Như ta đã thấy thì nhu cầu của khách hàng là vô tận nhưng khả năng đáp ứng là hữu hạn, vì vậy cần tập chung vào nhu cầu cấp thiết hiện nay của khách hàng. Có rất nhiều điểm giữ xe hiện nay nhưng chưa thực sự đáp ứng được những yêu cầu mà khách hàng mong muốn, nguyên nhân là do: mức độ an toàn của những điểm giữ xe trên không làm khách hàng hoàn toàn an tâm, thái độ phục vụ của những nhân viên gửi xe đối với khách hàng chưa được tế nhị, cũng như mức độ chuyên nghiệp, thời gian gửi xe làm cho khách hàng khó chịu vì phải đợi lâu mới tới lượt mình.

Như vấn đề bên trên đã đặt ra thì nhu cầu sử dụng bãi xe thông minh ở thị trường Việt Nam rất cao và thậm trí là lên đến đỉnh điểm. Nắm bắt được điểm yếu đó trên thị trường nên nhiều công ty về công nghệ đã không ngừng phát triển các hệ thống bãi xe thông minh nhằm đáp ứng nhu cầu của người dân cũng như là thu về lợi nhuận rất khủng.

Như ở các bãi xe truyền thống trước đó, bộ phận quẩn lý gặp rất nhiều khó khăn trong công việc quản lý và cũng không ít trường hợp nhân viên giữ xe bị gây khó dễ, việc thất thoát tài chính là điều không tránh khỏi, rồi ở một số nơi việc ghi phấn lên xe khiến nhiều người khó chịu, còn nếu ghi vé xe bằng giấy thì một số trường hợp bị ướt sẽ rất khó xử lý, . . . gây lãng phí và ô nhiễm môi trường.

Thị trường nước ngoài đã có rất nhiều mô hình hiện đại nhưng giá cả còn đắt đỏ,

và nước ta cũng hạn chế về lập trình.

Để khắc phục những vấn đề đó thì đề tài này phần nào góp phần giải quyết được nhu cầu về bãi xe, tích hợp được công nghệ RFID sử dụng thẻ từ vào để cải thiện tính hiệu quả và linh hoạt trong các hệ thống hiện có.

**1.3. MỤC ĐÍCH LUẬN ÁN**

Mục đích của đề tài là thiết kế và thi công được một hệ thống bãi gửi xe gồm mô hình một bãi gửi xe tự động và phầm mềm quản lý trên PC ứng dụng công nghệ RFID và xử lí ảnh, lưu trữ và xử lí dữ liệu trên phần mềm Excel.

**1.4. KẾT LUẬN CHƯƠNG I**

**1.4.1.Nội dung nghiên cứu**

- Xác định mục tiêu và giới hạn đề tài.

- Nghiên cứu tài liệu về Kit Arduino UNO R3.

- Thiết kế khối hiển thị trên LCD, khối chấp hành, khối nguồn, Reader và bộ xử lí trung tâm.

- Thiết kế, tính toán và thi công cho phần cứng.

- Viết code cho Kit Arduino Uno R3.

- Lắp rắp và kết nối giữa khối xử lí trung tâm và các khối khác.

- Hiển thị dữ liệu lên LCD.

- Chạy thử nghiệm hệ thống.

- Chỉnh sửa các lỗi điều khiển, lỗi lập trình và lỗi của các thiết bị.

- Viết báo cáo đồ án.

**1.4.2. Giới hạn**

Đề tài: “Bãi xe tự động sử dụng công nghệ RFID” có những giới hạn sau:

- Sử dụng Kit Arduino Uno R3.

- Sử dụng các nền tảng đã có sẵn và thư viện mở để phát triển sản phẩm.

- Sử dụng các module có sẵn như: module relay, module LCD I2C, MFRC522…

**1.4.3. Phạm vi sử dụng**

Đề tài là một mô hình nhỏ phục vụ cho việc nghiên cứu và phát triển trong quy mô phòng học và phòng thực tập.

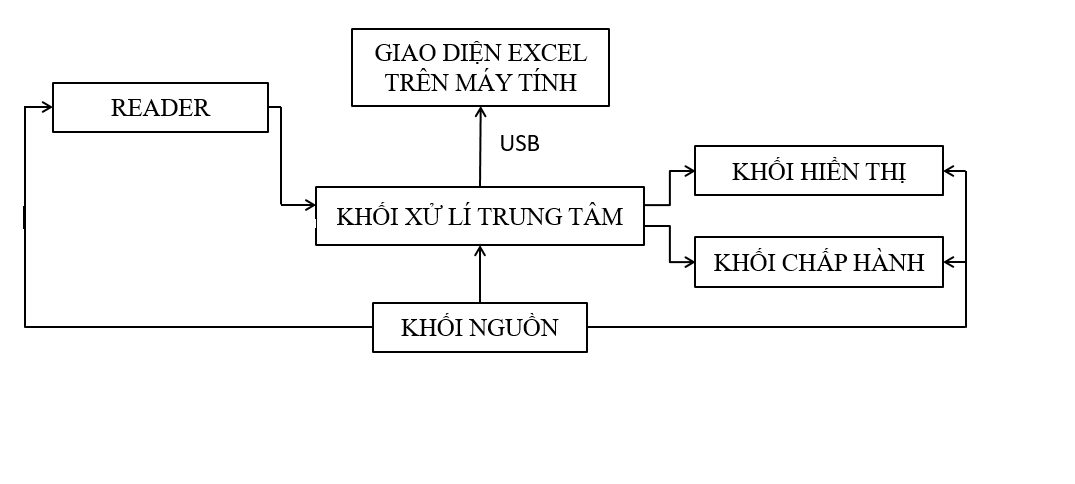
**Chương 2: THIẾT KẾ VÀ THỰC HIỆN CHƯƠNG TRÌNH**

**2.1. THIẾT KẾ TỔNG QUÁT**

**2.2.1. Yêu cầu của hệ thống**

* Hệ thống có các chức năng sau:
* Hiển thị LCD biển số xe ra, vào.
* Dùng thẻ RFID để kích hoạt đóng, mở cổng.
* Báo hiệu bằng còi, cửa tự mở, khi phát hiện thẻ.
* Bật đèn tự động khi có xe ra, vào.
* Lưu thời gian, thông tin người dùng, trạng thái ra vào bãi xe ở file Excel.

**2.2.2. Sơ đồ khối và chức năng mỗi khối**

****

Hình 2.1 : Sơ đồ khối của hệ thống

* Chức năng từng khối:
* **Khối xử lý trung tâm:** Truyền dữ liệu để cập nhật cơ sở dữ liệu hiển thị lên LCD, đồng thời truyền dữ liệu đến khối chấp hành và nhận tín hiệu từ khối đầu đọc reader, gửi dữ liệu lên máy tính.
* **Khối hiển thị:** Nhận dữ liệu từ khối xử lý trung tâm để hiển thị ra LCD biển số xe vào, ra bến.
* **Khối nguồn:** Cung cấp nguồn cho các khối: hiển thị LCD, chuông báo, trung tâm xử lý…
* **Khối chấp hành:** Nhận dữ liệu từ khối xử lý trung tâm, thực thi các yêu cầu được giao như servo, loa, đèn.
* **Khối Reader:** Đọc dữ liệu thừ thẻ RFID, truyền dữ liệu đến bộ xử lí trung tâm.
* **Giao diện Excel trên máy tính:** Nhận thông tin ID, tên, thời gian ra, vào từ khối xử lí trung tâm qua cổng USB.

**2.2. THIẾT KẾ CHI TIẾT**

**2.2.1. Phần cứng**

**2.2.1.1. Vi điều khiển**

* Giới thiệu về Arduino

Arduino là một board mạch vi xử lý dùng đê lập trình xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Ưu điềm của Arduino là ngôn ngữ cực kì dễ học (giống C/C++), cấp ngoại vi trên bo mạch đêu đã được chuẩn hóa, nên không cần biết nhiều về điện tử chúng ta cũng có thể lập trình được. Phân cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Những model hiện tại được trang bị gồm 1 công giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích vởi nhiêu board mở rộng khác nhau.

* Arduino Uno R3 SMD

|  |  |  |
| --- | --- | --- |
| IMG_256 |  | smdd |
| Hình 2.2: Hình ảnh thực tế Arduino UNO R3 SMD | Hình 2.3: Kí hiệu Ardunio UNO R3 SMD | Hình 2.4: Sơ đồ chân Arduino UNO R3 SMD |

* Thông số kỹ thuật

|  |  |  |
| --- | --- | --- |
| **STT** | **Thông số** | **Giá trị** |
| **1** | Điện áp hoạt động | 5V DC (chỉ được cấp qua cổng USB) |
| **2** | Tần số hoạt động | 16 MHz |
| **3** | Dòng tiêu thụ | 30 mA |
| **4** | Dòng tối đa trên mỗi chân I/O | 30 mA |
| **5** | Dòng ra tối đa (5V) | 500 mA |
| **6** | Dòng ra tối đa ( 3.3V) | 50 mA |
| **7** | Điện áp vào giới hạn | 6-20V DC |

Bảng 2. 1: Bảng thông số kĩ thuật Arduino Uno R3

* Nguồn sử dụng: Arduino có thể được cấp nguồn thông qua cổng USB hoặc cấp nguồn ngoài thông qua jack cắm 2.1mm, cũng có thể sử dụng 2 chân Vin và GND để cấp nguồn cho Arduino.
* Chức năng các chân:
* Chân 5V và chân 3.3V: các chân này dùng để lấy nguồn ra tương ứng 5V và 3.3V.
* Vin: Cấp nguồn cho Arduino.
* GND: chân nối mass.
* Reset: chân thiết lập lại hoạt động từ đầu cho board khi nhận tín hiệu.

**2.2.1.2. Công nghệ RFID và Module MFRC522**

* Giới thiệu công nghệ RFID

Công nghệ RFID là công nghệ nhận dạng đối tượng băng sóng vô tuyến. Công nghệ này cho phép nhận biết các đối tượng thông qua hệ thống thu phát sóng radio, từ đó có thể giám sát, quan lý từng đối tượng.

Một thiết bị hay một hệ thống RFID được cấu tạo bời hai thành phân chính là thiết bị đọc (reader) và thiết bị phát mã RFID có gắn chip. Thiết bị đọc được gắn anten để thu - phát sóng điện từ, thiết bị phát mã RFID được gắn với vật cản nhận dạng, mỗi thiết bị RFID chứa một mã số nhất định và không trùng lặp nhau.

* Các tần số thường được sử dụng trong hệ thống RFID:
* LF: 125 kHz – 34.2 kHz (low frequencies): ứng dụng nhiều cho hệ thống quan lý nhân sự, chấm công, cửa bào mật, bãi giữ xe...
* HF: 13.56 MHz (high frequencies): ứng dụng nhiều cho quản lý nguồn gốc hàng hóa, vận chuyển hàng hóa, cửa bảo mật, bãi giữ xe...
* UHF: 860 MHz - 960 MHz (ultra high frequencies): ứng dụng nhiều trong các hệ thống kiểm soát như thu phí đường bộ tự động, kiểm kê kho hàng, kiểm soát đường đi của hàng hóa...
* SHF: 2.45 GHz: (super high frequencies): ứng dụng nhiều trong các hệ thống kiểm soát như thu phí đường bộ tự động, kiếm soát lưu thông hàng hóa, kiểm soát hàng hóa, kiểm kê kho hàng...
* Module RFID MFRC522

|  |  |
| --- | --- |
|  |  |
| Hình 2. 5: Hình thực tế Module RFID RC522 Hình 2. 6:Kí hiệu MFRC522 | |

* Thông số kỹ thuật

|  |  |  |
| --- | --- | --- |
| **STT** | **Thông số** | **Giá trị** |
| **1** | Điện áp hoạt động | 3.3V DC |
| **2** | Tần số hoạt động | 13.56 MHz |
| **3** | Dòng tiêu thụ | 26 mA |
| **4** | Khoảng cách hoạt động | 0-60 mm |
| **5** | Tốc độ tối đa | 10 Mbps |

Bảng 2. 2: Bảng thông số kĩ thuật

* Chức năng các chân:
* SDA: kết nối với chân SPI\_SDA của vi điều khiển để lựa chọn chip khi giao tiếp SPI (Kích hoạt ờ mức thấp).
* SCK: Kết nối với chân SPI­\_SDA của vi điều khiển để tạo xung trong chế độ truyền SPI.
* MIS0: Kết nối với chân SPI \_M1SO của vi diêu khiển có chức năng Master Data Out- Slave In trong chế độ giao tiếp SPI.
* M0SI: Kết nối với chân SPI\_MOS1 của vi diều khiển có chức năng Master Data In- Slave Out trong chế độ giao tiêp SPI.
* IRQ: Chân ngắt.
* GND: Chân nối mass.
* RST: Chân Reset.
* VCC: Chân cấp nguồn.

**2.2.1.3. Giới thiệu LCD 16x2**

LCD (Liquid Crystal Display) được sử dụng trong rất nhiều các ứng dụng của Vi Điều Khiển. LCD có rất nhiều ưu điểm so với các dạng hiển thị khác. Nó có khả năng hiển thị kí tự đa dạng, trực quan (chữ, số và kí tự đồ họa), dễ dàng đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau, tốn rất ít tài nguyên hệ thống và giá thành rẻ.

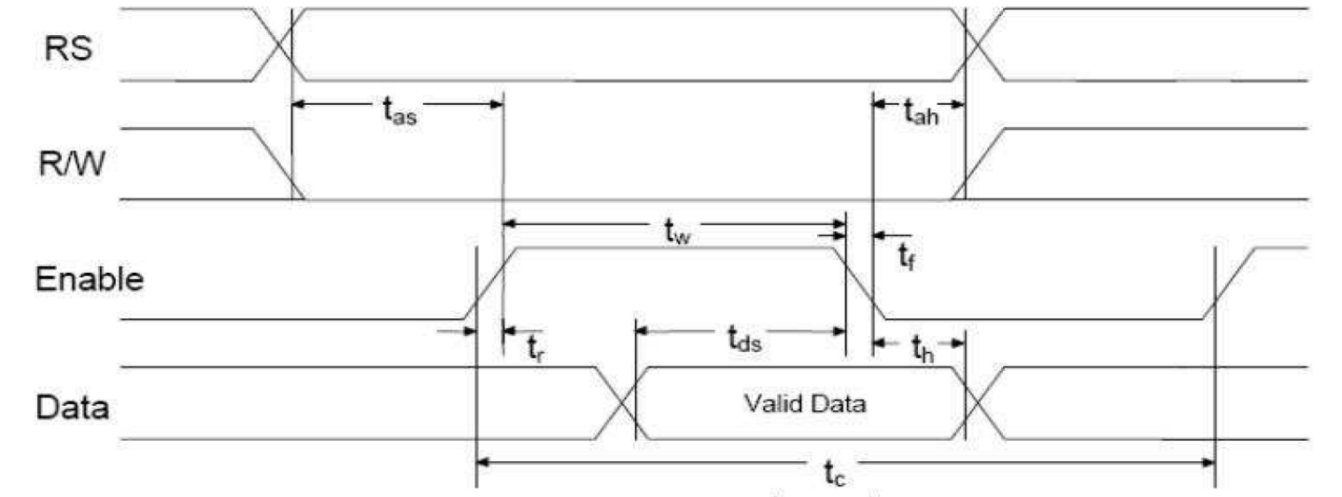
|  |  |
| --- | --- |
|  |  |
| **Hình 2. 7:** Hình ảnh thực tế LCD 16x2 | Hình 2. 8: Kí hiệu LCD 16x2 |

* Thông số kỹ thuật

|  |  |  |
| --- | --- | --- |
| **STT** | **Thông số** | **Giá trị** |
| **1** | Điện áp hoạt động | 2.5V ~ 6V |
| **2** | Dòng điện tối đa | 30mA |
| **3** | Nhiệt độ hoạt động | -30 ~ 75OC |
| **4** | Trọng lượng | 5g |

Bảng 2. 3: Thông số kĩ thuật LCD 16x2

* Cách điều khiển LCD:



Hình 2. 9: Sơ đồ điều khiển LCD 16x2

* Nhìn vào dạng sóng ta có thể thấy được trình tự điều khiển như sau:
* Điều khiển tín hiệu RS.
* Điều khiến tín hiệu R/W xuống mức thấp.
* Điêu khiên tín hiệu E lên mức cao đê cho phép.
* Xuất dữ liệu D7 - DO.
* Điều khiển tín hiệu E về mức thấp.
* Điều khiến tín hiệu R/W lên mức cao trở lại.
* Chức năng các chân:
* Chân 1: (Vss) Chân nối đất cho LCD.
* Chân 2: VDD Là chân cấp nguồn cho LCD.
* Chân 3: VEE là chân điều chỉnh độ tương phản của LCD.
* Chân 4: RS Là chân chọn thanh ghì (Register select). Nối chân RS với logic “0” (GND) hoặc logic “1” (VCC) để chọn thanh ghi.

+ Logic “0”: Bus DB0-DB7 sẽ nối với thanh ghi lệnh IR của LCD (ở chế độ ghi) hoặc nối với bộ đểm địa chỉ cùa LCD (ở chể độ đọc).

+ Logic “1”: Bus DB0-DB7 sẽ nối với thanh ghi dữ liệu DR bên trong LCD

* Chân 5: R/W là chân chọn chế độ đọc/ghi (Read/Wrìte). Nối chân R/W với logic “0” đề LCD hoạt động ớ chế độ ghi, hoặc nối với logic “1 ” để LCD ờ chế độ đọc.
* Chân 6: E Là chân cho phép (Enable). Sau khi các tín hiệu được đặt lên bus DB0-DB7, các lệnh chỉ được chấp nhận khi có 1 xung cho phép cua chân E.

+ Ởchế độ ghi: Dữ liệu ở bus sẽ dược LCD chuyên vào (chấp nhận) thanh ghi bên trong nó khi phát hiện một xung (high-to-low transition) của tín hiệu chân E.

+ Ởchế độ đọc: Dữ liệu sẽ được LCD xuất ra DB0-DB7 khi phát hiện cạnh lên (low-to-high transition) ờ chân E và được LCD giữ ở bus đến khi nào chân E xuống mức thấp.

* Chân 7 - 14: DBO - DB7 tám đường của bus dữ liệu dùng để trao đôi thông tin với MPU. Có 2 chế dộ sử dụng 8 dường bus này:

+ Chế độ 8 bít: Dữ liệu được truyền trên cà 8 đường, với bit MSB là bit DB7.

+ Chế độ 4 bít: Dữ liệu được truyền trên 4 đường từ DB4 tới DB7, bit MSB là DB7.

* Chân 15: Nguồn dương cho đèn nền.
* Chân 16: GND cho đèn nền.

**2.2.1.4. Module I2C**

Bằng việc sử dụng giao tiếp I2C, việc điều khiển trực tiếp màn hình được chuyển sang cho IC xử lý nằm trên mạch. Chỉ cần việc gửi mã lệnh cùng nội dung hiển thị, do vậy giúp vi điều khiển có nhiều thời gian xử lý các tiến trình phức tạp khác.

I2C sử dụng hai đường truyền tín hiệu: một đường xung nhịp đồng hồ (SCL) và một đường tín hiệu (SDA). SCL và SDA luôn được kéo lên nguồn bằng một điện trở kéo lên có giá trị xấp xỉ 4.7 KΩ.

Các chế độ hoạt động cùa I2C bao gồm:

* Chế độ chuẩn (standard mode) hoạt động ở tốc độ 100 Kbit/s.
* Chế độ tần số thấp (low speed mode) hoạt động ớ tốc độ 10 Kbit/s.

Ngoài ra I2C còn có chế độ 10 bit đia chỉ tương đương với 1024 địa chỉ.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Hình 2. 10: Hình thực tế module I2C | Hình 2. 11: Sơ đồ các chân I2C | Hình 2.12: Kí hiệu I2C |

* Thông số kỹ thuật:

|  |  |  |
| --- | --- | --- |
| **STT** | **Thông số** | **Giá trị** |
| **1** | Điện áp hoạt động | 2.5-6VDC |
| **2** | Trọng lượng | 5g |
| **3** | Kích thước | 41.5mm(L)X19mm(W)X15.3mm(H) |

Bảng 2. 4: Thông số kĩ thuật Module I2C

* Ưu điểm của việc sử dụng giao tiếp I2C
* Giao tiếp I2C chỉ sử dụng duy nhất 2 dây tín hiệu: SDA và SCL giúp tiết kiệm chân trên vi điều khiển.
* Tốc độ truyền dữ liệu lên đến 400Kbps.
* Dữ liệu truyền nhận đảm bảo tính toàn vẹn vì sử dụng cơ chế phản hồi (ACK) trên mỗi byte dữ liệu.
* Có khả năng kết nối nhiều thiết bị với nhau: trên mạch có sẵn các mối hàn.

A0, A1, A2 để thay đối địa chỉ của module.

**2.2.1.5. Buzzer**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Hình 2.13: Hình thực tế buzzer | Hình 2.14: Sơ đồ các chân buzzer | Hình 2.15: Kí hiệu buzzer |

* Thông số kỹ thuật:

|  |  |  |
| --- | --- | --- |
| **STT** | **Thông số** | **Giá trị** |
| **1** | Điện áp | 3.5-5.5VDC |
| **2** | Dòng tiêu thụ | <25mA |
| **3** | Tần số cộng hưởng | 2300Hz ± 500Hz |
| **4** | Biên độ âm thanh | >80dB |

Bảng 2. 5: Thông số kĩ thuật buzzer

**2.2.1.6. Động cơ Servo SG90**

|  |  |  |
| --- | --- | --- |
| Hình 2.16: Hình thực tế động cơ Servo SG90 | Buy Tower Pro SG90 Servo Motor Online – QuartzComponents  Hình 2.17: Sơ đồ các chân động cơ Servo SG90 | Hình 2.18: Kí hiệu động cơ Servo SG90 |

* Thông số kỹ thuật:

|  |  |  |
| --- | --- | --- |
| **STT** | **Thông số** | **Giá trị** |
| **1** | Điện áp | 4.8~5.2 VDC |
| **2** | Dòng tiêu thụ | <25mA |
| **3** | Tốc độ hoạt động | 60O trong 0.12s |
| **4** | Góc quay tối đa | 0~180O |
| **5** | Kích thước | 40,7 x19,7 x 42.9 (mm) |

Bảng 2. 6: Thông số kĩ thuật động cơ Servo

* Chức năng các chân
* Kết nối dây màu đỏ (VCC) với 5V.
* Dây màu nâu (GND) với mass.
* Dây màu cam (PWM) với chân phát xung của vi điều khiển.

**2.2.2. Lập trình**

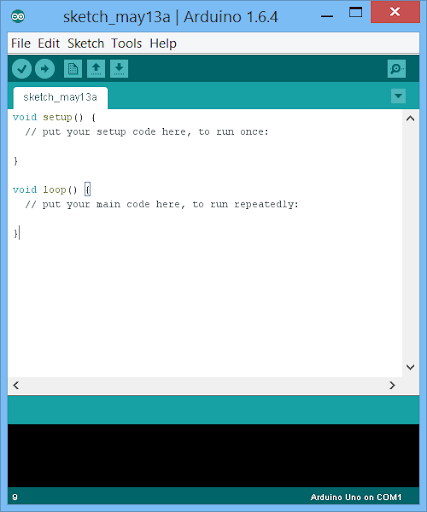
**2.2.2.1. Phần mềm Arduino IDE**

Với mục tiêu của đề tài là sử dụng Arduino Uno R3 nên phần mềm để viết chương trình điều khiển là phần mềm Arduino IDE.

Arduino IDE (Arduino Integrated Development Environment) là một trình soạn thảo văn bản. giúp viết code để nạp vào bo mạch arduino. Arduino có thể chạy trên Windows, MAC OS và Linux.

Một chương trình viết bởi Arduino IDE được gọi là sketch, sketch được lưu dưới định dạng .ino.

Sau khi khơi động Arđuino IDE, ta được giao diện như hình sau:



Hình 2. 19:Giao diện phần mềm ARDUINO IDE

**2.2.2.2. Phần mềm Microsoft Excel**

Microsoft Excel là một ứng dụng bảng tính thuộc bộ Microsoft Office, công dụng chính của nó là giúp người dùng tính toán nhanh và chính xác với số lượng dữ liệu lớn. Excel là một tập hợp các hàng và cột được kết hợp với nhau tạo ra một bảng tính lớn. Các hàng thường được ký hiệu lần lượt bằng các chữ cái như: A, B, C, D, E,… còn các cột thì được ký hiệu bằng các con số từ 1, 2, 3 cho đến n…. Một cột và một hàng gộp lại với nhau sẽ trở thành một ô tính. Địa chỉ của ô sẽ được xác định bởi chữ cái đại diện cho cột và số đại diện cho hàng.

Cấu trúc của Excel:

* Bảng tính – Sheet: Sheet là bảng tính trong Excel. Mỗi bảng tính có tới hơn 4 triệu ô dữ liệu, người dùng có thể thoải mái sử dụng chúng để tính toán, lập biểu mẫu, phân tích dữ liệu…
* Sổ tay – Workbook: một Workbook thường có từ 1 – 255 sheet riêng biệt. Workbook thường được sử dụng để tập hợp những loại bảng tính, đồ thị… có sự liên hệ với nhau. Excel không giới hạn số lượng Workbook nhưng việc mở được bao nhiêu Workbook sẽ tùy thuộc vào bộ nhớ máy tính.
* Cột – Column: Cột là tập hợp của các ô tính theo chiều dọc. Độ rộng mặc định của mỗi cột  là 9 ký tự (có thể thay đổi trị số này từ 0 – 255 ký tự). Một bảng tính Excel có tới 256 cột, mỗi cột được gán một chữ cái khác nhau như: A, B, C, E, F, G, H…
* Dòng – Row: Dòng là tập hợp của các ô tính theo chiều ngang. Chiều cao mặc định của dòng là là 12.75 chấm điểm (có thể thay đổi trị số này từ 0 – 409). Một bảng tính Excel có ít nhất 16384 dòng, mỗi dòng được gán một chữ số như: 1, 2, 3, 4, 5…
* Ô – Cell: Ô là điểm giao nhau của dòng và cột. Tọa độ của một ô được xác định bởi  ký hiệu số thứ tự của dòng và chữ cái ký hiệu của cột.
* Vùng – Range: Vùng thường là tập hợp của nhiều ô lân cận nhau, nó thường được xác định bởi toạ độ ô đầu và toạ độ ô cuối. Vùng có thể là một ô, một nhóm ô hay là toàn bộ bảng tính, có thể làm việc với nhiều vùng khác nhau cùng một lúc.

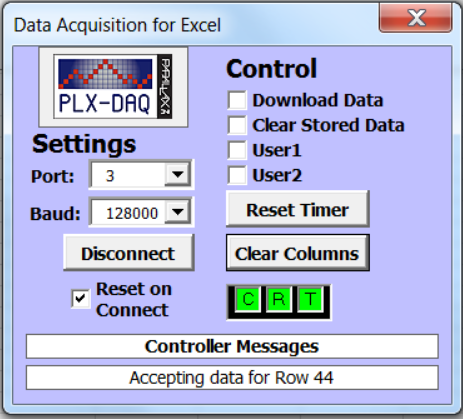
**2.2.2.3. Công cụ Parallax Data Acquisltion**

PLX-DAQ là một công cụ bổ trợ thu thập dữ liệu vi điều khiển Parallax dành cho Microsoft Excel. Bất kỳ bộ vi điều khiển nào của chúng tôi được kết nối với bất kỳ cảm biến nào và cổng nối tiếp của PC hiện có thể gửi dữ liệu trực tiếp vào Excel. Phần mềm bổ trợ thu thập tới 26 kênh dữ liệu từ bất kỳ bộ vi điều khiển thị sai nào và đưa các số vào cột khi chúng đến. PLX-DAQ cung cấp khả năng phân tích bảng tính dễ dàng đối với dữ liệu được thu thập tại hiện trường, phân tích các cảm biến trong phòng thí nghiệm và giám sát thiết bị theo thời gian thực.

PLX-DAQ có các tính năng sau:

* Dữ liệu đồ thị hoặc đồ thị khi nó đến trong thời gian thực bằng Microsoft Excel
* Ghi lên đến 26 cột dữ liệu
* Đánh dấu dữ liệu bằng thời gian thực (hh: mm: ss) hoặc giây kể từ khi đặt lại
* Đọc / Ghi bất kỳ ô nào trên trang tính
* Đọc / Đặt bất kỳ trong 4 hộp kiểm trên điều khiển giao diện
* Tốc độ truyền lên đến 128K Hỗ trợ Com1-15 yêu cầu hệ thống

Ở trang tính Excel với PLX-DAQ đã được kích hoạt, cửa sổ Data Acquisation for Excel như hình dưới đây sẽ xuất hiện:



Hình 2.20: Giao diện kết nối giữa Arduino và Excel

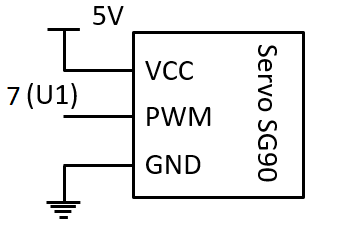
Cần thực hiện lựa chọn cổng giao tiếp (Port) và tốc độ truyền (Baud) tương ứng với kết nối của arduino, sau đó kết nối (Connect) để PLX-DAQ bắt đầu thực hiện việc nhận dữ liệu.

Khi hoàn tất, dữ liệu sẽ được cập nhật vào các Cell tương ứng do Arduino định trước trong Sheet đầu tiên của file Excel. Việc hiển thị dữ liệu này lên các dạng đồ thị hoặc sơ đồ là khá đơn giản, tùy theo ý đồ và trình Excel của người sử dụng.

* 1. **TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG PHẦN CỨNG** 
     1. **Khối chấp hành:**

**2.3.1.1. Động cơ Servo SG-90**

Động cơ RC Servo 90 là động phổ biến dùng trong các mô hình điều khiển nhỏ và đơn giản như cánh tay robot hay các cơ cấu chấp hành dơn giản. Động cơ có tốc độ phản ứng nhanh, được tích hợp sẵn driver điều khiển động cơ, dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM.

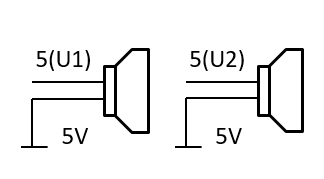
****

Hình 2.21: Kết nối động cơ Servo SG-9G với Arduino Uno

* Kết nối cảm động cơ với Ardunio:
* Chân VCC:  kết nối nguồn 5V.
* Chân GND:  mass chung.
* Chân PWM: kết nối với chân số 7 của board Arduino UNO R3.
* Mô tả hoạt động:
* Động cơ RC servo được điều khiển bằng cách cấp một chuỗi xung PWM. Tần số của xung điều khiển nên là 50 Hz, góc quay của động cơ phụ thuộc vào độ rộng của xung điều khiển. Loại động cơ này có giới hạn về góc quay, góc quay tối đa là 180 độ.
* Khi có tín hiệu PWM từ Ardunio gửi đến thì Servo thực hiện xoay động cơ đóng/mở cửa.

**2.3.1.2. Buzzer**

Buzzer là một thiết bị tạo ra tiếng còi hoặc tiếng bíp. Có nhiều loại nhưng cơ bản nhất là buzzer áp điện, là một miếng phẳng của vật liệu áp điện với hai điện cực. Loại buzzer này đòi hỏi phải có các bộ dao động để điều khiển nó. Loại buzzer này tuổi thọ cao, hiệu suất ổn định, chất lượng tốt, được sản xuất nhỏ gọn phù hợp thiết kế với các mạch còi buzzer nhỏ gọn, mạch báo động.

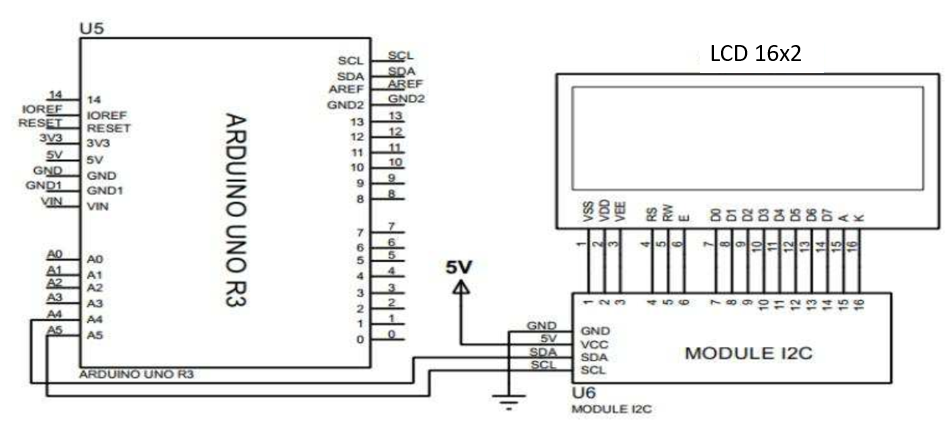


Hình 2.22: Kết nối buzzer (loa) với Arduino Uno

* Kết nối loa với Ardunio:
* Chân dương (+) nối với VCC
* Chân âm (-) nối với chân tín hiệu số 8 của board Arduino UNO R3.
* Mô tả hoạt động: Nguyên tắc hoạt động rất đơn giản là sẽ phát ra âm thanh khi chân tín hiệu được bật sang LOW và sẽ tắt khi chân tín hiệu bật về HIGH.
  + 1. **Khối hiển thị: LCD\_I2C**

LCD có quá nhiều nhiều chân gây khó khăn trong quá trình đấu nối và chiếm dụng nhiều chân trên vi điều khiển. Module I2C giải quyết vấn đề giảm số chân kết nối, thay vì mất 6 chân vi điều khiển để kết nối với LCD 16x2 thì module I2C chỉ cần tốn 2 chân (SCL,SDA) để kết nối. Module I2C hỗ trợ các loại LCD sử dụng driver HD44780 (LCD 16x2, 20x4...) và tương thích với hầu hết các dòng vi diều khiển hiện nay. Vì vậy em sử dụng module này để thực hiện đề tài.

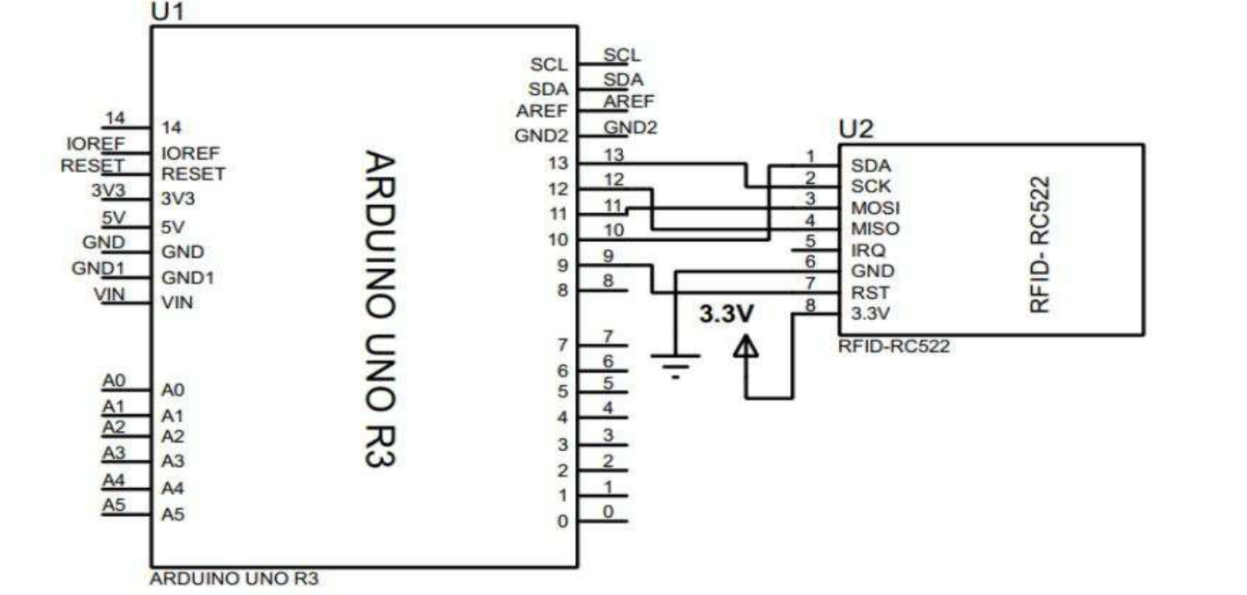
Sơ đồ kết nối khối hiển thị với khối xử lí trung tâm:



Hình 2.23: Sơ đồ kết nối khối hiển thị với khối xử lí trung tâm

* Kết nối module I2C LCD với Ardunio:
* Chân VCC:  kết nối nguồn 5V.
* Chân GND:  đất chung.
* Chân SDA: kết nối với chân A4 của board Arduino.
* Chân SCL: kết nối với chân số A5 của board Arduino.
* Mô tả hoạt động: Khối hiển thị được kết nối với Ardunio Uno và nhận dữ liệu sau nó hiển thị thông tin tình trạng xe ra, vào.
  + 1. **Khối Reader**

Trên thị trường hiện nay có nhiều loại đầu đọc thẻ RFID với nhiều giá thành và chất lượng khác nhau, vì yêu cầu của hệ thống đơn giản và sự phù hợp của module với board xử lí, dễ phát triển, chỉ cần đọc, ghi, phát hiện thẻ RFID nên nhóm đã chọn module MFRC522 để phù hợp với hệ thống.

****

Hình 2.24 : Kết nối khối Reader (MFRC522) với Arduino Uno

* Kết nối với Ardunio:

|  |  |  |
| --- | --- | --- |
| **STT** | **RFID RC522** | **ARDUINO UNO R3** |
| 1 | 3.3V | 3.3V |
| 2 | RST | PIN 9 |
| 3 | GND | GND |
| 4 | MISO | PIN11 |
| 5 | MOSI | PIN12 |
| 6 | SCK | PIN13 |
| 7 | SDA | PIN10 |

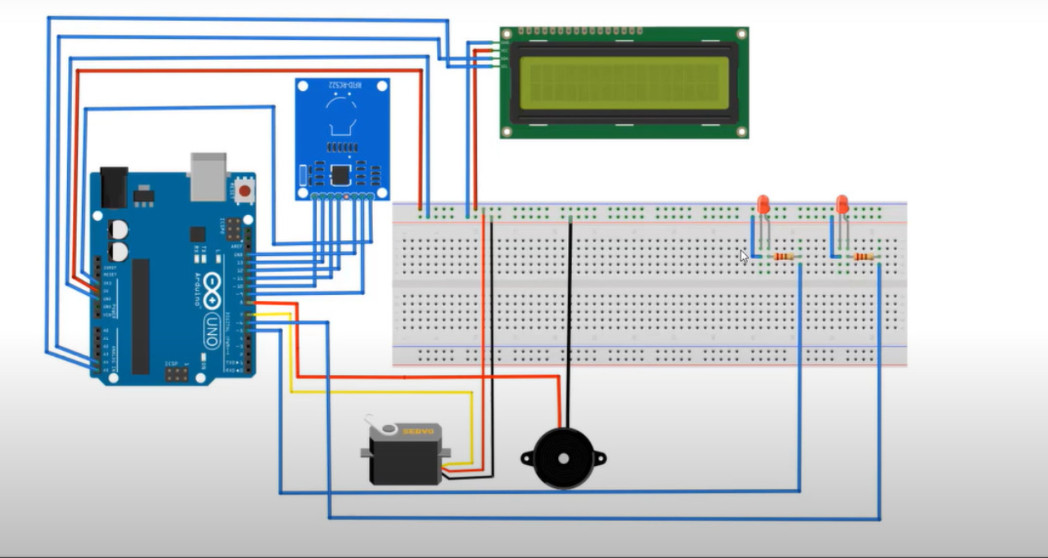
Bảng 2.7 : Sơ đồ kết nối chân MFRC522 với Arduino

* Mô tả hoạt động: Khi có thẻ được quét vào module MFRC522 thì dữ liệu ID của thẻ được đọc bởi bộ đọc và gửi về khối xử lí trung tâm sau đó so sánh với mã ID dữ liệu trong code và tiếp tục thực thi.
  + 1. **Khối xử lý trung tâm:**

Trên thị trường hiện nay có nhiều Board nhúng như: ARM, Raspberry, Arduino, PIC…với nhiều tính năng khác nhau. Với sự thông dụng và gần gủi, Arduino được các bạn sinh viên chọn để học tập và nghiên cứu. Em cũng đã chọn Board Arduino Uno R3 để làm khối xử lí trung tâm cho hệ thống.

Khối xử lí trung tâm có sơ đồ kết nối sau:

Từ các mô tả trên, ta có sơ đồ nguyên lí toàn mạch như hình:



Hình 2.25 : Sơ đồ nguyên lí toàn mạch

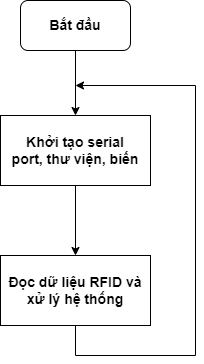
**2.3.5. Khối nguồn**

|  |  |  |
| --- | --- | --- |
| Thiết bị | Dòng (mA) | Điện áp(V) |
| Arduino UNO R3 | 30x2 | 5 |
| LCD I2C | 35 x 2 | 5 |
| Module MFRC522 | 26 x 2 | 3.3 |
| Động cơ Servo SG90 | 10 x 2 | 5 |
| Buzzer | 15 x 2 | 5 |

Bảng 2.8 : Dòng và áp của các linh kiện được sử dụng trong hệ thống

Từ các yêu cầu về nguồn của hệ thống gồm nguồn cung cấp 5V, 3.3V và yêu cầu về dòng điện tiêu thụ khoảng 0.5A. Nhóm đã chọn adapter 5V-1A để làm nguồn cung cấp cho hệ thống, giúp hệ thống ổn định, nguồn 3.3V được sử dụng từ vi điều khiển.

* 1. **LƯU ĐỒ GIẢI THUẬT HỆ THỐNG**
     1. **Lưu đồ giải thuật chương trình chính**



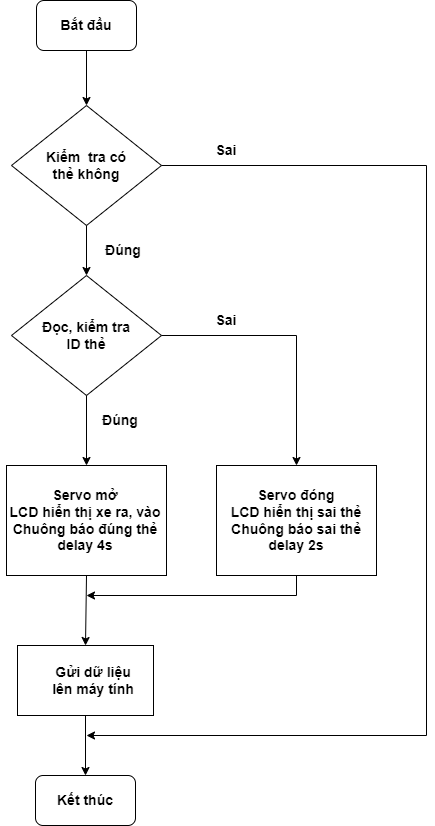
Giải thích lưu đồ

- Đầu tiên khởi tạo các cổng kết nối UART, khởi tạo các thư viện các biến sử dụng trong chương trình.

- Tiếp theo thực hiện chương trình phát hiện, đọc thẻ RFID.

Hình 2.26: Lưu đồ giải thuật chương trình chính

* + 1. **Lưu đồ chương trình con RFID**



Giải thích lưu đồ

- Đầu tiên, kiểm tra có thẻ quét vào đầu đọc hay không. Nếu không thì kết thúc chương trình, nếu có thì thực hiện bước kế tiếp.

- Sau đó thực hiện đọc dữ liệu ID từ đầu đọc gửi về khối xử lí trung tâm, so sánh với ID lưu trong code:

+ Nếu đúng: Mở cửa, LCD hiển thị xe ra vào chuông báo thẻ đúng kêu, giữ trạng thái 4 giây.

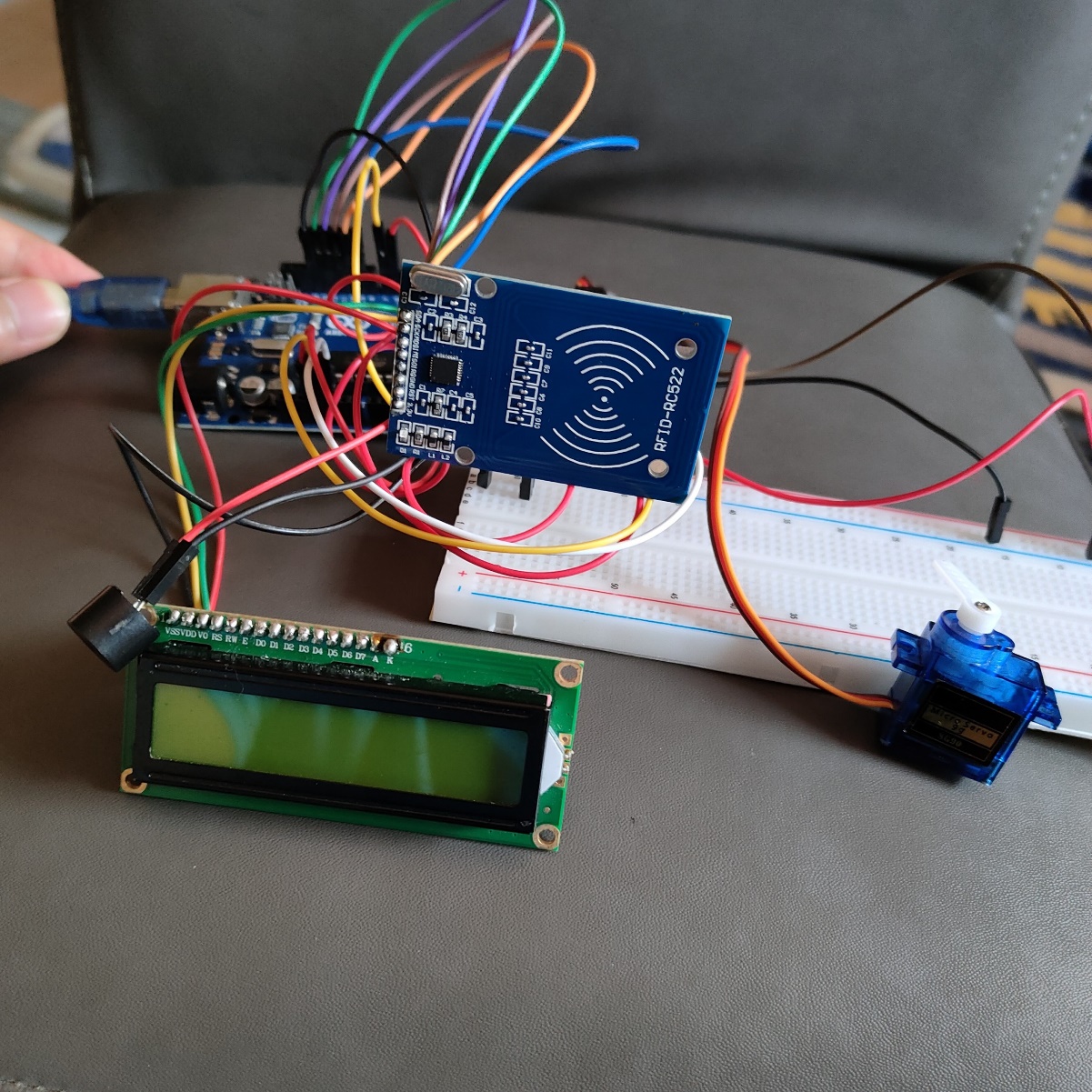
+ Nếu sai: Đóng cửa, LCD hiển thị báo sai thẻ, chông báo thẻ sai kêu, giữ trạng thái 2 giây.

- Tiếp theo sẽ gửi dữ liệu ngày giờ, ID thẻ, tên người dùng, trạng thái ra vào lên máy tính.

Hình 2.27: Lưu đồ chương trình con RFID

**2.5. SẢN PHẨM HOÀN THIỆN**

**2.5.1. Mô hình RFID**

****

**Hình 2.28. Mô hình RFID**

**+** Kích thước : 30 x 30 cm

+ Điện áp làm việc và công suất: cắm trực tiếp với máy tính thông qua cổng COM

+ Nhiệt độ và ánh sáng: tránh để ánh nắng trực tiếp quá lâu.

**2.5.2. Chương trình Arduino IDE**

+ Hệ điều hành: đã có trên tất cả hệ điều hành

+ Giao diện: Tiếng Anh và Tiếng Việt

+ Dung lượng cài đặt: 530MB

+ Chạy code, kết nối với mạch Arduino qua cổng COM

+ Truyền dữ liệu qua Excel thông qua PLX-DAQ

**2.5.3. Phần mềm Excel và công cụ PLX-DAQ**

+ Hệ điều hành: đã có trên tất cả hệ điều hành

+ Giao diện: Tiếng Anh và Tiếng Việt

+ Dung lượng cài đặt: 100 Mbs

+ Hiện thị dữ liệu từ mạch Arduino

+ Cập nhật khi truyền dữ liệu từ Arduino vào

**2.6. KẾT LUẬN CHƯƠNG II**

+ Đầu tiên, chúng ta sẽ lắp mạch theo sơ đồ đấu nối ở **phần 2.3**

**+** Tiếp theo chúng ta sẽ kết nối cổng COM với mạch và nạp phần code trong Arduino IDE

+ Tiếp đến chúng ta kết nối công cụ PLX-DAQ với Arduino thông qua nhận diện cổng COM để hiển thị dữ liệu

+ Bước cuối cùng là quét các thẻ từ đã chuẩn bị trước và xem thành quả.

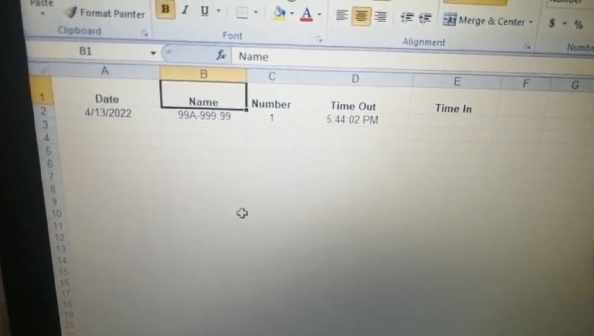
**Chương 3: THỬ NGHIỆM MẠCH THỰC VÀ KẾT QUẢ**

**3.1. KỊCH BẢN THỬ NGHIỆM**

**Chương trình máy tính:**

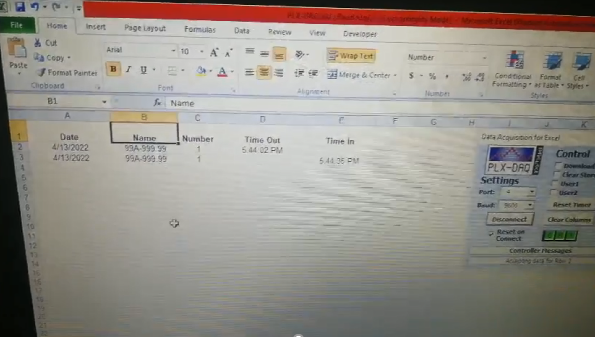
- Khởi tạo: Xuất dữ liệu trắng lên Excel

- Chức năng xuất dữ liệu lên Excel khi xe vào bến:



Hình 3. 1: Dữ liệu xe vào bến

-Chức năng xuất dữ liệu lên Excel khi xe ra bến:

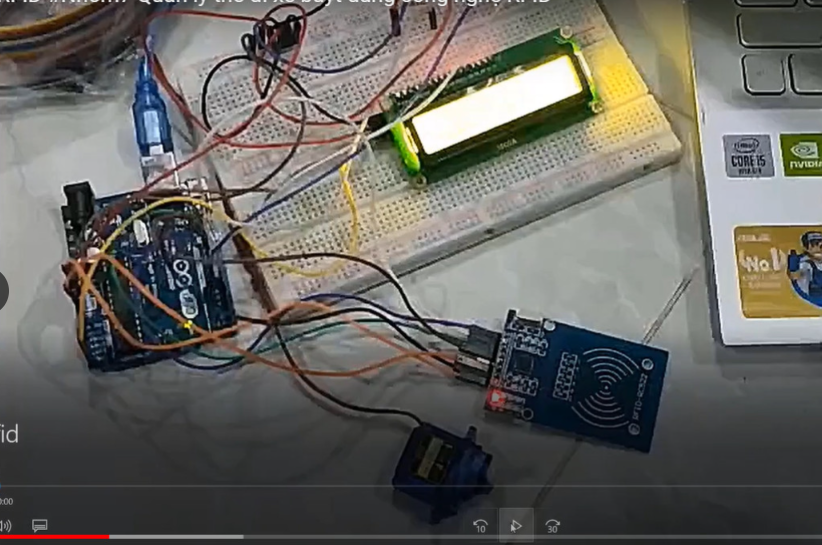


Hình 3. 2: Dữ liệu xe ra bến

-Khi lỗi : Thông tin sẽ không hiển thị lên Excel

**3.2. KẾT QUẢ THỰC NGHIỆM**

Sau quá trình nghiên cứu thiết kế, lập trình và kiểm thử em đã hoàn thành được mô hình đề tài, do tình hình dịch bệnh hiện tại và điều kiện đi lại chưa cho phép nên em chỉ hoàn thành đồ án ở dạng testboard và vẫn còn một số thiếu sót về thiết bị.



Hình 3. 3: Mô hình mạch testboard của đề tài

Khi khởi tạo bắt đầu hệ thống màn hình LCD ở ngõ vào sẽ hiển thị “SYSTEM READY SCAN YOUR CARD ”



Hình 3. 4: trạng thái LCD

Khi quét thẻ RFID vào đầu đọc thẻ ở ngõ vào, đầu đọc sẽ gửi mã ID của thẻ về bộ xử lí trung tâm để kiểm tra, nếu ID thẻ đúng:

* Đúng: LCD sẽ hiển thị: ID xe vào bến. Thanh barie sẽ mở, chuông báo thẻ đúng sẽ kêu và led xanh sáng, sau 4 giây sẽ trở lại trạng thái lúc khởi tạo.
* Sai: LCD sẽ hiển thị “THE KHONG HOP LE XIN THU LAI”. Thanh barie sẽ đóng, chuông báo thẻ sai sẽ kêu và led đỏ sáng, sau 4 giây sẽ trở lại trạng thái lúc khởi tạo ban đầu.

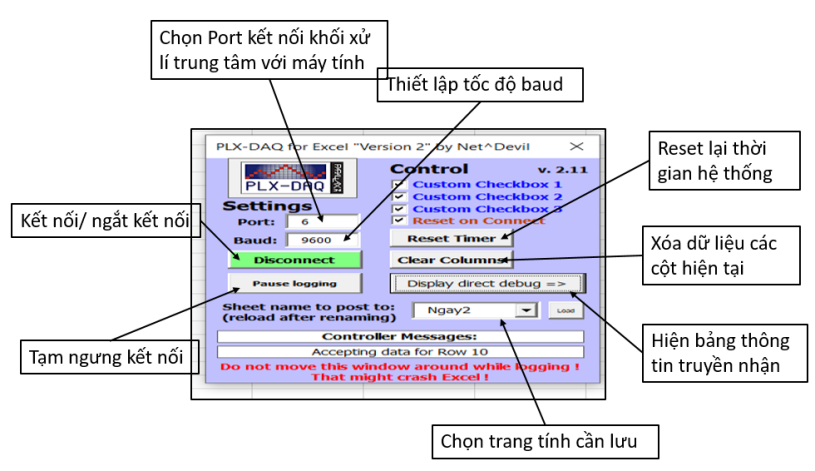
|  |  |
| --- | --- |
|  |  |
| Hình 3. 5: Trạng thái LCD khi quét ID thẻ đúng ở ngõ vào | Hình 3. 6: Trạng thái LCD khi quét ID thẻ sai ở ngõ vào |

Khi quét thẻ RFID vào đầu đọc thẻ ở ngõ ra, đầu đọc sẽ gửi mã ID của thẻ về bộ xử lí trung tâm để kiểm tra, nếu ID thẻ:

* Đúng: LCD sẽ hiển thị ID xe ra bến. Thanh barie sẽ mở, chuông báo thẻ đúng sẽ kêu, sau 4 giây sẽ trở lại trạng thái lúc khởi tạo.
* Sai: LCD sẽ hiển thị “THE KHONG HOP LE XIN THU LAI”. Thanh barie sẽ đóng, chuông báo thẻ sai sẽ kêu và led đỏ sáng, sau 4 giây sẽ trở lại trạng thái lúc khởi tạo ban đầu.

|  |  |
| --- | --- |
|  |  |
| Hình 3. 7: Trạng thái LCD khi quét ID thẻ đúng ở ngõ ra | Hình 3. 8: Trạng thái LCD khi quét ID thẻ sai ở ngõ ra |

Mở file Excel đã cài đặt công cụ PLX-DAQ,chọn kết nối với cổng COM máy tính, thiết lập tốc độ baund, chọn trang tính lưu thông tin, sau đó bấm Connect để kết nối, ngoài ra có thể thiết lập thêm một số thuộc tính phụ cho trang tính.



Hình 3. 9: Bảng thiết lập kết nối giữa bộ xử lí trung tâm và file Excel trên máy tính

Thông tin ngày, giờ, mã ID thẻ, tên người dùng, trạng thái vào/ ra sẽ được hiển thị trên trên file Excel. Đối với thẻ đúng đã (được lưu trong hệ thống) sẽ hiện tên người dùng, đối với thẻ sai (chưa có trong hệ thống) sẽ hiển thị “Warning. Wrong Card” ở mục USER.

**3.3. ĐÁNH GIÁ KẾT QUẢ**

Sau khi hoàn thiện ta tiến hành chạy thử toàn bộ hệ thống để kiểm tra độ ổn định, cũng như tính chính xác và độ trễ khi điều khiển hệ thống. Sản phẩm đáp ứng được những tiêu chí đề ra trong quá trình lên ý tưởng và thiết kế sản phẩm, các thiết bị và module hoạt động tốt. Tuy nhiên trong một thời gian dài chạy kiểm thử thì hệ thống vẫn chưa ổn định, dễ bị nhiễu nguồn.

**3.4. KẾT LUẬN CHƯƠNG III**

Đã thực hiện được hầu hết các mục tiêu đề ra về việc hiển thị dữ liệu lên phần mềm Excel, chuông và đèn led ở phần cứng hoạt động, LCD hiện thị được kết quả. Kết quả sát với thử nghiệm. Trong quá trình thực nghiệm em rút ra cần phải điều chỉnh khoảng cách nhận dạng, thời gian chuông kêu và đèn báo cho phù hợp hơn khi đưa vào thực tế.

**Chương 4: KẾT LUẬN CHUNG VÀ HƯỚNG PHÁT TRIỂN**

**4.1. KẾT LUẬN CHUNG**

Sau thời gian thực hiện để tài đến nay em đã hoàn thành được các yêu cầu đề ra, đã thiết kế và kiểm thử thành công mô hình bãi giữ xe ứng dụng công nghệ RFID với những phương pháp và chức năng như sau:

- Lập trình Arduino để đọc dữ liệu từ RFID, nhận dữ liệu về gửi dữ liệu lên file Excel, hiển thị LCD.

- Sử dụng các chuẩn và phương thức giao tiếp thường gặp là UART, SPI, I2C.

- Ứng dụng công nghệ RFID để đọc dữ liệu ID từ thẻ.

- Điều khiển động cơ servo để mở thanh barie cho xe ra vào.

Một số ưu điểm của hệ thống:

- Giải quyết cơ bản về vấn đề kẹt xe, mất vé gửi xe, đánh tráo thẻ xe trong các bãi xe truyền thống trước đó.

- Giảm rất nhiều thời gian, các khoản chi phí thuê nhân viên quản lý, bảo vệ, thu về được lợi nhuận cao hơn. Khả năng đọc và ghi lại được dữ liệu nhiều lần. (khi khách hàng trả đổi thẻ hoặc không sử dụng nữa có thể tái sử dụng lại thẻ với khách hàng mới).

- Thẻ RFID rất bền, có thể hoạt động tốt trong các môi tnường không thuận lợi như nóng, ẩm, bụi bắn, ăn mòn, hay có sự va chạm.... tạo được độ tin cậy cao khi sử dụng.

Tuy nhiên do kiến thức còn hạn hẹp nên không thể tránh khỏi những thiếu sót và vẫn còn hạn chế về hệ thống. Một số hạn chế của hệ thống:

- Chi phí triển khai cao hơn các bãi xe truyền thống trước đó.

- Thẻ RFID dễ bị nhiễu sóng trong môi trường nước và kim loại.

- Việc thêm, xóa, tái chế thẻ cần phải thực hiện ở phần code điều khiển.

- Dễ bị nhiễu nguồn, lỗi hiển thị LCD khi hoạt động lâu hay reset lại hệ thống.

**4.2. HƯỚNG PHÁT TRIỂN**

Để mô hình có thể phát triển trong tương lai nhóm có một số phương án đề xuất sau:

- Sử dụng các linh kiện có độ chính xác cao để nâng tính ổn định và đảm bảo cho hệ thống.

- Kết hợp thêm xử lí ảnh, lập trình thêm chức năng nhận đạng người lái xe, biển số xe cho hệ thống, tăng cường thêm tính bảo mật.

- Kết hợp thêm các tấm pin năng lượng mặt trời đặt ở bãi xe, hướng tới mục tiêu sử dụng nguồn năng lượng sạch và tiết kiệm.

- Cập nhật dữ liệu lên cloud để tránh trường hợp bị mất điện hay hệ thống xảy ra sự cố ngoài ý muốn.

**TÀI LIỆU THAM KHẢO**

[1] Nguyễn Đăng Việt, Trần Trí Đạt, Đồ án tốt nghiệp: “Thiết kế, thi công bãi giữ xe ứng dụng công nghệ RFID và xử lí ảnh”, 2019.

[2] Quản lí bãi xe thông minh: https://www.youtube.com/watch?v=H3VJM7b4XkM