

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC THĂNG LONG**

-----□□&□□-----



**CHUYÊN ĐỀ TỐT NGHIỆP**

**Xây dựng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ**

**GIÁO VIÊN HƯỚNG DẪN: TS. NGUYỄN NGỌC TÂN**

**SINH VIÊN THỰC HIỆN: A**

**A33779\_VÕ TIẾN ĐẠT**

**A40850\_V****Ũ THỊ CHIẾN**

**NGÀNH: MẠNG MÁY TÍNH VÀ TRUYỀN**

**THÔNG DỮ LIỆU**

**HÀ NỘI – 2023**

**LỜI CẢM ƠN**

Lời đầu tiên, chúng em xin được gửi lời cảm ơn chân thành nhất tới toàn thể các thầy giáo, cô giáo trong chuyên ngành Mạng máy tính và truyền thông dữ liệu, Khoa Công nghệ thông tin cũng như các thầy cô giảng dạy trong trường Đại học Thăng Long đã truyền đạt những kiến thức quý báu và bổ ích nhất cho chúng em trong những năm học vừa qua.

Đặc biệt, chúng em xin gửi lời cảm ơn chân thành và sâu sắc tới thầy – TS. Nguyễn Ngọc Tân, giảng viên khoa Công Nghệ Thông Tin, Đại học Thăng Long. Thời gian qua nhờ vào sự hướng dẫn tận tình và sự động viên, giúp đỡ ân cần từ thầy mà chúng em mới có thành quả là xây dựng và hoàn thiện được Chuyên đề tốt nghiệp như ngày hôm nay.

Bên cạnh đó, chúng em cũng xin chân thành gửi lời cảm ơn đến tất cả những người bạn cùng khoá, những người anh, người chị khoá trên đã nhiệt tình giúp đỡ và chia sẻ cho chúng em những kinh nghiệm và kiến thức đáng quý trong suốt quá trình chúng em xây dựng dự án của mình.

Vì kiến thức còn nhiều hạn chế, trong quá trình triển khai và thực hiện Chuyên đề tốt nghiệp này chúng em không tránh khỏi có những sai sót không mong muốn, chính vì vậy chúng em rất mong nhận được những ý kiến đóng góp từ các thầy cô cũng như các bạn trong khoa để hoàn thiện dự án này cũng như bổ sung những kiến thức còn thiếu sót.

Cuối cùng chúng em xin kính gửi lời chúc tới toàn thể các thầy cô trong khoa Toán - Tin cũng như các thầy cô đang giảng dạy trong trường Đại học Thăng Long luôn luôn mạnh khỏe và tràn đầy nhiệt huyết để truyền đạt kiến thức cho các thế hệ mai sau.

Chúng em xin chân thành cảm ơn!

Hà Nội, 30 tháng 06 năm 2023

# MỤC LỤC

[MỤC LỤC 2](#_Toc139120689)

[DANH MỤC HÌNH ẢNH 4](#_Toc139120690)

[DANH MỤC TỪ VIẾT TẮT 5](#_Toc139120691)

[LỜI GIỚI THIỆU 6](#_Toc139120692)

[CHƯƠNG 1. TỔNG QUAN HỆ THỐNG 8](#_Toc139120693)

[1.1. GIỚI THIỆU ĐỀ TÀI 8](#_Toc139120694)

[1.2. MỤC TIÊU, PHƯƠNG PHÁP, NHIỆM VỤ VÀ Ý NGHĨA CỦA ĐỀ TÀI 9](#_Toc139120695)

[1.2.1. Mục tiêu 9](#_Toc139120696)

[1.2.2. Phương pháp nghiên cứu 9](#_Toc139120697)

[1.2.3. Nhiệm vụ của đề tài. 9](#_Toc139120698)

[1.2.4. Ý nghĩa của đề tài 9](#_Toc139120699)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 10](#_Toc139120700)

[2.1. Giới thiệu về phần cứng 10](#_Toc139120701)

[2.1.1. Kit Arduino Mega 2560 10](#_Toc139120702)

[2.1.2. LCD 16x2 13](#_Toc139120703)

[2.1.3. Led đơn 15](#_Toc139120704)

[2.1.4. Cảm biến vật cản hồng ngoại 16](#_Toc139120705)

[2.2. Giới thiệu về phần mềm 18](#_Toc139120706)

[2.2.1. Giới thiệu về Arduino 18](#_Toc139120707)

[2.2.2. Phần mềm mô phỏng Proteus 19](#_Toc139120708)

[CHƯƠNG 3. Phân tích và thiết kế hệ thống 22](#_Toc139120709)

[3.1. Phân tích 22](#_Toc139120710)

[Sơ đồ khối 22](#_Toc139120711)

[3.2. Thuật toán giải quyết bài toán tắc đường 23](#_Toc139120712)

[3.2.1. Sơ đồ thuật toán giải quyết tắc đường: 26](#_Toc139120713)

[3.2.2. Thuật toán đếm lưu lượng xe của cảm biến 27](#_Toc139120714)

[3.2.3. Thuật toán kiểm tra tắc đường 28](#_Toc139120715)

[3.2.4. Thuật toán xác định làn đường bị tắc 28](#_Toc139120716)

[3.3. Các bài toán tắc đường 29](#_Toc139120717)

[3.3.1. Kịch bản 1: Tắc nghẽn 1 chiều tại ngã tư 29](#_Toc139120718)

[3.3.2. Kịch bản 2: Tắc nghẽn 2 chiều tại ngã tư 30](#_Toc139120719)

[3.3.3. Kịch bản 3: Tắc nghẽn 3 chiều tại ngã tư 31](#_Toc139120720)

[3.3.4. Kịch bản 4:Tắc nghẽn tại tất cả 4 chiều của ngã tư 31](#_Toc139120721)

[3.4. Nguyên lý hoạt động 32](#_Toc139120722)

[3.5. Thiết kế hệ thống trên Proteus 33](#_Toc139120723)

[CHƯƠNG 4. Triển khai và đánh giá kết quả 35](#_Toc139120724)

[4.1. Triển khai các thông số mô phỏng: 35](#_Toc139120725)

# DANH MỤC HÌNH ẢNH

[Hình 2.1. Sơ đồ và chức năng chân của Arduino Mega 2560 10](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120858)

[Hình 2.2. Sơ đồ giao tiếp UART 12](#_Toc139120859)

[Hình 2.3. Màn hình LCD 16 x 2 13](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120860)

[Hình 2.4. Module i2C LCD 16×2 14](#_Toc139120861)

[Hình 2.5. Sơ đồ đấu nối giao tiếp iC2 với LCD 16×2. 15](#_Toc139120862)

[Hình 2.6. Led đơn sắc 15](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120863)

[Hình 2.7. Sơ đồ chân led đơn sắc 16](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120864)

[Hình 2.8. Cấu tạo của module cảm biến hồng ngoại Arduino 16](#_Toc139120865)

[Hình 2.9. Sơ đồ mô tả nguyên lý hoạt động của cảm biến hồng ngoại 17](#_Toc139120866)

[Hình 2.10. Sơ đồ kết nối của cảm biến hồng ngoại 18](#_Toc139120867)

[Hình 2.11. Giao diện chính trên proteus 8 20](#_Toc139120868)

[Hình 3.1. Sơ đồ khối kết nối của hệ thống 22](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120869)

[Hình 3.2. Sơ đồ minh hoạ hệ thống 23](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120870)

[Hình 3.3. Sơ đồ mô phỏng hệ thống 23](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120871)

[Hình 3.4. Sơ đồ thuật toán giải quyết bài toán đặt ra 25](#_Toc139120872)

[Hình 3.5. Thuật toán 1: Sơ đồ giải phóng làn đường bị tắc 26](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120873)

[Hình 3.6. Thuật toán 2: Sơ đồ thuật toán đếm lưu lượng xe 27](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120874)

[Hình 3.7. Thuật toán 3: Sơ đồ thuật toán kiểm tra tắc đường 28](#_Toc139120875)

[Hình 3.8. Mô phỏng mạch trên phần mềm Proteus 33](#_Toc139120876)

[Hình 4.1. Sơ đồ minh hoạ hệ thống 35](https://thanglongedu-my.sharepoint.com/personal/a40850_thanglong_edu_vn/Documents/6_Smart_Traffic_Management_System_%20_.docx#_Toc139120877)

# DANH MỤC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Ý nghĩa** |
| **1** | **CĐTN** | Chuyên đề tốt nghiệp |
| **2** | **VDC** | Volt direct current |
| **3** | **UART** | Universal Asynchronous Receiver / Transmitter |
| **4** | **PWM** | Pulse Width Modulation |
| **5** | **LCD** | Liquid Crystal Display |
| **6** | **IDE** | Integrated Development Environment |

# LỜI GIỚI THIỆU

Sự phát triển nhanh chóng của nền kinh tế và lượng dân cư đông đúc. Nhu cầu đi lại, vận chuyển cũng tăng lên nhanh chóng đi kèm theo nó là bài toán tắc nghẽn ùn tắc giao thông. Ùn tắc giao thông gây thiệt hại không nhỏ cho sự phát triển kinh tế quốc gia, giảm hiệu suất lao động và tăng các chi phí không cần thiết trong quá trình sản xuất. Trong bối cảnh kinh tế lạm phát và khó khăn như hiện nay, lãng phí trong giao thông lại đặt thêm một gánh nặng đối với đời sống kinh tế của người dân. Nguyên nhân của vấn đề này một phần là do cơ sở hạ tầng chưa đáp ứng được nhu cầu lưu thông hiện nay, một phần là do việc phân chia, định thời gian của hệ thống đèn giao thông hiện tại ở các giao lộ chưa hợp lý khiến cho số lượng phương tiện giao thông bị ùn tắc theo thời gian rồi dẫn đến tắc nghẽn giao thông. Qua thực trạng đỏ, nhóm muốn xây dựng một hệ thống điều tiết giao thông tự động, phân luồng trực tiếp và hiệu quả hơn phần nào đó giảm bớt tình trạng ùn tắc giao thông. Đồng thời tiếp cận, bổ sung các kiến thức mới, cũng như củng cố lại những kỹ năng kiến thức trong suốt quá trình học tập tại trường.

Do đó, nhóm em thực hiện đồ án với đề tài: “Xây dựng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ”.

Nghiên cứu này nhằm mục tiêu phát triển giải thuật điều khiển hệ thống đèn giao thông cho một giao lộ, với chu kỳ đèn tín hiệu tùy thuộc vào tình trạng xe lưu thông trên hai tuyến đường quan sát được bởi các cảm biến hồng ngoại. Giải thuật đếm số lưu lượng xe trên đường được xây dựng để quyết định thời gian của chu kỳ đèn xanh kế tiếp cho tuyến đường tương ứng. Số lượng xe thu được từ cảm biến sẽ được phần mềm mô phỏng giả lập để kiểm chứng giải thuật điều khiển. Kết quả mô phỏng cho thấy tuyến đường nào có lưu lượng xe lớn hơn thì chu kỳ đèn xanh sẽ dài hơn.

Nội dung của tài liệu bao gồm các chương sau đây:

**Chương 1**: Tổng quan hệ thống

Trình bày các khái niệm, định nghĩa và mô hình hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ. Đồng thời xác định mục tiêu nghiên cứu trong CĐTN.

**Chương 2**: Cơ sở lý thuyết

Chi tiết về bài toán đề ra “Xây dựng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ” trong CĐTN.

**Chương 3**: Phân tích và thiết kế hệ thống

Tiến hành nghiên cứu, chọn lọc ra những thiết bị cần thiết. Từ đó, thiết kế môi trường triển khai mô hình “Xây dựng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ.”

**Chương 4**: Kết quả thực nghiệm

Thực hiện triển khai, kiểm tra kết quả và đánh giá sản phẩm .

# TỔNG QUAN HỆ THỐNG

## GIỚI THIỆU ĐỀ TÀI

Ùn tắc giao thông trên mạng lưới đường bộ không gì khác ngoài tốc độ chậm hơn, thời gian chuyến đi tăng lên và hàng đợi của các phương tiện gia tăng. Khi số lượng phương tiện vượt quá sức chứa của đường, tình trạng tắc nghẽn giao thông xảy ra. Ở các thành phố đô thị của Việt Nam, tắc nghẽn giao thông là một vấn đề lớn. Ùn tắc giao thông xảy ra khi nhu cầu vượt quá khả năng cung cấp công suất đường bộ. Các sự cố cá nhân như tai nạn hoặc xe phanh đột ngột trong một luồng giao thông đông đúc trôi chảy có hiệu ứng gợn sóng và gây tắc đường. Thậm chí có những vấn đề nghiêm trọng về an ninh trật tự trong hệ thống đèn giao thông do các thành phần chống đối xã hội cũng dẫn đến tình trạng ùn ứ giao thông tại một số địa điểm. Sự tăng trưởng kinh tế chắc chắn đã có tác động đến giao thông đô thị. Khi thu nhập tăng lên, ngày càng có nhiều người bắt đầu đi ô tô hơn là xe hai bánh. Do đó, cần phải quản lý giao thông một cách thông minh vì việc quản lý giao thông theo cách thông thường như hệ thống tín hiệu không có tác dụng lớn trong việc hạn chế tắc nghẽn giao thông.

Để giải quyết tất cả những vấn đề này, chúng em đã lựa chọn đề tài: “Xây dựng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ”, một hệ thống quản lý lưulượng thời gian thực, năng động dựa trên cảm biến hồng ngoại và phân tích đã được đềxuất. Một trong những mục tiêu chính của hệ thống là luôn lưu thông, **t**hông suốt. Điều này có nghĩa là tắc nghẽn giao thông phải được ngănchặn trước khi nó xảy ra, thay vì quản lý nó sau khi nó xảy ra. Bằngcách sử dụng phân tích, nếu tắc nghẽn giao thông được dự đoán ngay cảsau khi kiểm soát hẹn giờ dựa trên mật độ, bảng điều khiển được sửdụng để hiển thị thông báo về độ lệch bắt buộc của lưu lượng truy cậptại các nút giao thông trước đó. Điều này đảm bảo tránh tắc nghẽngiao thông ngay cả trước khi nó xảy ra. Có thể đảm bảo cho các phươngtiện khẩn cấp đi qua bằng cách bật tín hiệu màu xanh lá cây cho lànđường phát hiện có phương tiện khẩn cấp. Khả năng của hệ thống vượtra ngoài quản lý giao thông. Các cảm biến hồng ngoại được gắn trên mặt đường tại các ngã tư. Sự cố phương tiện, tai nạn đường bộ và vi phạm giaothông như nhảy đèn đỏ có thể được phát hiện ngay lập tức và thông báocho các dịch vụ khẩn cấp và sở cảnh sát giao thông tương ứng.

## MỤC TIÊU, PHƯƠNG PHÁP, NHIỆM VỤ VÀ Ý NGHĨA CỦA ĐỀ TÀI

### Mục tiêu

+ Tìm hiểu về hệ thống đèn giao thông thông minh

+ Đưa ra những giải pháp giải quyết bài toán tắc đường cục bộ.

### Phương pháp nghiên cứu

+ Khai thác tài liệu hiện có về giao thông thông minh hiện nay.

+ Sử dụng những ứng dụng đã được thực hiện và thành công ở trong nước và trên thế giới.

+ Tiến hành thử nghiệm trong phạm vi nhỏ có thể thực hiện được rồi dần dần áp dụng rộng rãi.

### Nhiệm vụ của đề tài.

+ Nghiên cứu tổng quan về hệ thống đèn giao thông thông minh.

+ Nghiên cứu giải quyết bài toán tắc đường cục bộ.

+ Nghiên cứu những ứng dụng của hệ thống trên thế giới.

+ Tìm hiểu những ứng dụng của hệ thống ở Việt Nam hiện nay và khả năng áp dụng trong tương lai đã góp phần gì cho việc cải thiện tình hình giao thông.

### Ý nghĩa của đề tài

Với tình hình giao thông hiện nay, sử dụng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ là một trong những đòi hỏi cấp thiết cần được quan tâm, tìm hiểu kỹ lưỡng.

Hệ thống đèn giao thông thông minh là một trong những hệ thống an toàn và cần thiết cho chúng ta hiện nay.

Tại Việt Nam, hệ thống đèn giao thông thông minh đã xuất hiện từ khá lâu nhưng còn nhỏ lẻ và chưa hiệu quả.

Vì tất cả những lý do trên mà nhóm sinh viên chúng em đã chọn đề tài nghiên cứu: " Xây dựng hệ thống đèn giao thông thông minh và giải quyết bài toán tắc đường cục bộ”.

# CƠ SỞ LÝ THUYẾT

## Giới thiệu về phần cứng

### Kit Arduino Mega 2560

A picture containing text, diagram, plan, electronic engineering

Description automatically generatedTrong mô hình này, những người thực hiện dùng Board Arduino Mega 2560 được tích hợp sẵn Vi điều khiển Atmega2560 cho số ngoại vi, các chuẩn giao tiếp và số chân nhiều nhất, bộ nhớ rất lớn (256KB), dễ dàng kết nối với Module Arduino Motor Shield, nếu có những ứng dụng cần mở rộng thêm nhiều chân, nhiều ngoại vi thì đây là một sự lựa chọn đáng giá, board hoàn toàn có cấu trúc chân tương thích với các board như Uno và chạy điện áp 5VDC. Tất cả các Shield của Arduino Uno đều chạy được với Arduino Mega và Adruino Mega không dùng được thư viện SoftwareSerial vì đã có sẵn 4 bộ UART[4].

Hình .. Sơ đồ và chức năng chân của Arduino Mega 2560

Một vài thông số của Arduino Mega:

* Arduino Mega sử dụng vi điều khiển Atmega2560.
* Điện áp hoạt động: 5V – DC
* Tần số hoạt động: 16 Mhz
* Dòng điện áp: 30 mA
* Điện áp giới hạn: 7-9V – DC
* Số chân Analog: 16
* Số chân Digital: 54 (15 chân PWM)
* Giao tiếp UART: 4 bộ UART
* Giao tiếp SPI: 1 bộ (chân 50 -> 53) dùng với thư viện SPI của Arduino.
* Giao tiếp i2C: 1 bộ
* Ngắt ngoài: 6 chân
* Dòng tối đa trên mỗi chân i/O: 30 mA

Các chân năng lượng:

* GND (Ground): cực âm của nguồn điện cấp cho Arduino Mega.
* 5V: cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA
* 3.3V: cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA
* Vin (Voltage input): để cấp nguồn ngoài cho Arduino Mega.

Bộ nhớ của Arduino Mega 2560:

* 256kB bộ nhớ Flash: những đoạn lệnh lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển. Sẽ có khoảng vài KB trong số này sẽ được dùng cho botloader nhưng thường thì ít khi nào phải sử dụng quá 20kb bộ nhớ này.
* 8Kb cho SRAM (Static Random Access Memory): giá trị các biến khai báo khi lập trình sẽ lưu ở đây. Khai báo càng nhiều biến thì càng cần nhiều bộ nhớ RAM.
* Chú ý: khi mất điện, dữ liệu trên SRAM sẽ bị mất.
* 4KB cho EEPROM (Electrically Eraseble Programmable Read Only Memory): tương tự như một chiếc ổ cứng mini-nơi có thể đọc và ghi dữ liệu vào đây mà không phải lo bị mất khi mất điện giống như lưu ở trên SRAM.

UART là gì?

UART hay bộ thu-phát không đồng bộ đa năng là một trong những hình thức giao tiếp kỹ thuật số giữa thiết bị với thiết bị đơn giản và lâu đời nhất. Bạn có thể tìm thấy các thiết bị UART trong một phần của mạch tích hợp (IC) hoặc dưới dạng các thành phần riêng lẻ. Các UART giao tiếp giữa hai nút riêng biệt bằng cách sử dụng một cặp dẫn và một nối đất chung.

A picture containing text, line, screenshot, diagram

Description automatically generated

Hình .. Sơ đồ giao tiếp UART

Hướng dẫn giao tiếp UART

Vì nó là thiết lập phổ quát nên chúng ta có thể định cấu hình UART để hoạt động với nhiều loại giao thức nối tiếp khác nhau. UART đã được điều chỉnh thành các đơn vị chip đơn vào đầu những năm 1970, bắt đầu với Western Digital’s WD1402A.

Trong một sơ đồ giao tiếp UART:

Chân Tx (truyền) của một chip kết nối trực tiếp với chân Rx (nhận) của chip kia và ngược lại. Thông thường, quá trình truyền sẽ diễn ra ở 3.3V hoặc 5V. UART là một giao thức một master, một slave, trong đó một thiết bị được thiết lập để giao tiếp với duy nhất một thiết bị khác.

Dữ liệu truyền đến và đi từ UART song song với thiết bị điều khiển (ví dụ: CPU).

Khi gửi trên chân Tx, UART đầu tiên sẽ dịch thông tin song song này thành nối tiếp và truyền đến thiết bị nhận.

UART thứ hai nhận dữ liệu này trên chân Rx của nó và biến đổi nó trở lại thành song song để giao tiếp với thiết bị điều khiển của nó.

UART truyền dữ liệu nối tiếp, theo một trong ba chế độ:

- Full duplex: Giao tiếp đồng thời đến và đi từ mỗi master và slave

- Half duplex: Dữ liệu đi theo một hướng tại một thời điểm

- Simplex: Chỉ giao tiếp một chiều

Dữ liệu truyền qua UART được tổ chức thành các gói. Mỗi gói chứa 1 bit bắt đầu, 5 đến 9 bit dữ liệu (tùy thuộc vào UART), một bit chẵn lẻ tùy chọn và 1 hoặc 2 bit dừng.

### man hinh lcd 16x2LCD 16x2

Hình .. Màn hình LCD 16 x 2

**Thông số kỹ thuật LCD 16×2**

**LCD 16×2** được sử dụng để hiển thị trạng thái hoặc các thông số.

LCD 16×2 có 16 chân trong đó 8 chân dữ liệu (D0 – D7) và 3 chân điều khiển (RS, RW, EN).

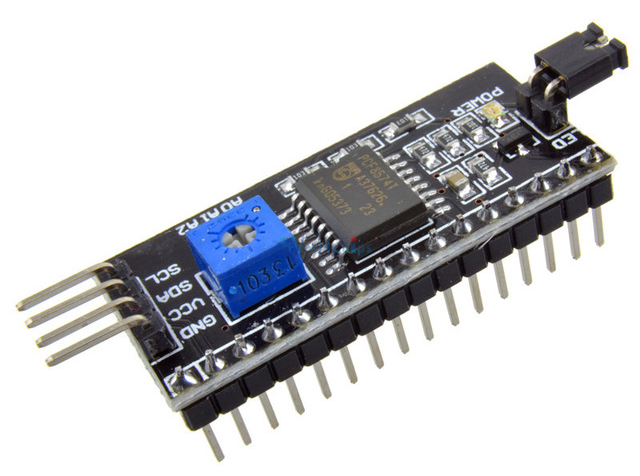
5 chân còn lại dùng để cấp nguồn và đèn nền cho LCD 16×2.

Các chân điều khiển giúp ta dễ dàng cấu hình LCD ở chế độ lệnh hoặc chế độ dữ liệu.

Chúng còn giúp ta cấu hình ở chế độ đọc hoặc ghi.

LCD 16×2 có thể sử dụng ở chế độ 4 bit hoặc 8 bit tùy theo ứng dụng ta đang làm.

**Module i2C Arduino**



Hình 2.4. Module i2C LCD 16×2

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 LCD ra đời và giải quyết vấn để này cho bạn.

Thay vì phải mất 6 chân vi điều khiển để kết nối với LCD 16×2 (RS, EN, D7, D6, D5 và D4) thì module iC2 bạn chỉ cần tốn 2 chân (SCL, SDA) để kết nối.

Module i2C hỗ trợ các loại LCD sử dụng driver HD44780(LCD 16×2, LCD 20×4, …) và tương thích với hầu hết các vi điều khiển hiện nay.

Thông số kĩ thuật

* Điện áp hoạt động: 2.5-6V DC.
* Hỗ trợ màn hình: LCD1602,1604,2004 (driver HD44780).
* Giao tiếp: i2C.
* Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2).
* Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt.
* Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD.

Giao tiếp i2C LCD Arduino:

Module i2C LCD 16×2 Arduino UNO

GND GND

VCC 5V

SDA A4/SDA

SCL A5/SCL

Sơ đồ đấu nối

A picture containing electronics, electronic component, circuit component, electronic engineering

Description automatically generated

Hình .. Sơ đồ đấu nối giao tiếp I2C với LCD 16×2.

### Led đơn

Gốm 3 led có màu khác nhau: xanh, vàng, đỏ để mô phỏng đèn tín hiệu giao thông.

Đối với led màu đỏ và vàng: VF = 1.9 – 2.2V, iF = 15 – 25 mA

A picture containing tube

Description automatically generatedĐối với led màu xanh: VF = 3.0 – 3.4V, iF = 15 – 25 mA

Hình .. Led đơn sắc

A diagram of anode and anode

Description automatically generated with low confidence

Hình .. Sơ đồ chân led đơn sắc

### Cảm biến vật cản hồng ngoại

Cảm biến vật cản hồng ngoại (Infrared Obstacle Sensor) là một loại cảm biến sử dụng công nghệ hồng ngoại để phát hiện và đo khoảng cách vật cản. Cảm biến này sử dụng tia hồng ngoại để phát ra một tín hiệu và đo thời gian mà tín hiệu này trở lại sau khi va chạm với vật cản, từ đó tính toán khoảng cách giữa cảm biến và vật cản.

Cảm biến phát hiện vật cản thường được sử dụng trong các ứng dụng như robot tự động hóa, xe tự lái, hệ thống giám sát khoảng cách, và hệ thống an ninh tự động. Nó có thể hoạt động ở mức năng suất cao, đồng thời cũng đảm bảo tính an toàn trong các ứng dụng tự động.

Cấu tạo của module cảm biến hồng ngoại Arduino:

A picture containing screenshot, diagram, clock, text

Description automatically generated

Hình .. Cấu tạo của module cảm biến hồng ngoại Arduino

* Chân VCC: là chân cấp nguồn cho module, thường được nối với nguồn 5V DC.
* Chân GND: là chân đất, được nối với chân đất của nguồn.
* Chân OUT: là chân dữ liệu, sẽ đưa ra tín hiệu khi có sự phát hiện của đối tượng.

Nguyên lý hoạt động của cảm biến vật cản hồng ngoại:

Cảm biến vật cản hồng ngoại hoạt động dựa trên nguyên lý thu phát sóng hồng ngoại. Khi cảm biến nhận được tín hiệu từ một nguồn phát sóng hồng ngoại, nó sẽ phản hồi lại tín hiệu đó và đo lường thời gian phản hồi của sóng để xác định khoảng cách từ cảm biến đến vật thể.

A picture containing sketch, diagram, design, antenna

Description automatically generated

Hình .. Sơ đồ mô tả nguyên lý hoạt động của cảm biến hồng ngoại

Cụ thể, cảm biến sử dụng một bộ phát sóng hồng ngoại để phát ra tín hiệu sóng hồng ngoại. Khi sóng hồng ngoại va chạm với vật thể, nó sẽ bị phản xạ trở lại và được nhận bởi một bộ thu sóng hồng ngoại trên cảm biến. Cảm biến sau đó sử dụng độ trễ thời gian giữa tín hiệu phát và tín hiệu nhận để tính toán khoảng cách từ cảm biến đến vật thể.

Sơ đồ đấu nối

A picture containing screenshot, display, rectangle, text

Description automatically generated

Hình .. Sơ đồ kết nối của cảm biến hồng ngoại

## Giới thiệu về phần mềm

### Giới thiệu về Arduino

1. **Phần mềm Arduino iDE là gì**

Arduino iDE là một phần mềm mã nguồn mở chủ yếu được sử dụng để viết và biên dịch mã vào module Arduino.

Đây là một phần mềm Arduino chính thức, giúp cho việc biên dịch mã trở nên dễ dàng mà ngay cả một người bình thường không có kiến thức kỹ thuật cũng có thể làm được.

Nó có các phiên bản cho các hệ điều hành như MAC, Windows, Linux và chạy trên nền tảng Java đi kèm với các chức năng và lệnh có sẵn đóng vai trò quan trọng để gỡ lỗi, chỉnh sửa và biên dịch mã trong môi trường.

Có rất nhiều các module Arduino như Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro và nhiều module khác.

Mỗi module chứa một bộ vi điều khiển trên bo mạch được lập trình và chấp nhận thông tin dưới dạng mã.

Mã chính, còn được gọi là sketch, được tạo trên nền tảng iDE sẽ tạo ra một file Hex, sau đó được chuyển và tải lên trong bộ điều khiển trên bo.

Môi trường iDE chủ yếu chứa hai phần cơ bản: Trình chỉnh sửa và Trình biên dịch, phần đầu sử dụng để viết mã được yêu cầu và phần sau được sử dụng để biên dịch và tải mã lên module Arduino.

Môi trường này hỗ trợ cả ngôn ngữ C và C++.

1. **Arduino iDE hoạt động như thế nào**

Khi người dùng viết mã và biên dịch, iDE sẽ tạo file Hex cho mã. File Hex là các file thập phân Hexa được Arduino hiểu và sau đó được gửi đến bo mạch bằng cáp USB. Mỗi bo Arduino đều được tích hợp một bộ vi điều khiển, bộ vi điều khiển sẽ nhận file hex và chạy theo mã được viết.

### Phần mềm mô phỏng Proteus

Proteus là phần mềm cho phép mô phỏng hoạt động của mạch điện tử bao gồm phần thiết kế mạch và viết chương trình điều khiển cho các họ vi điều khiển như MCS-51, PIC, AVR…

Proteus là phần mềm mô phỏng mạch điện tử của Lancenter Electronics, mô phỏng cho hầu hết các linh kiện điện tử thông dụng, đặc biệt hỗ trợ cho cả các MCU như PIC, 8051, AVR, Motorola.

Phần mềm bao gồm 2 chương trình: iSIS cho phép mô phỏng mạch và ARES dùng để vẽ mạch in. Proteus là công cụ mô phỏng cho các loại vi điều khiển khá tốt, nó hỗ trợ các dòng VĐK PIC, 8051, PIC, dsPIC, AVR, HC11, MSP430, ARM7/LPC2000... các giao tiếp i2C, SPI, CAN, USB, Ethenet... Ngoài ra còn mô phỏng các mạch số, mạch tương tự một cách hiệu quả. Proteus là bộ công cụ chuyên về mô phỏng mạch điện tử.

ISIS đã được nghiên cứu và phát triển trong hơn 12 năm và có hơn 12000 người dùng trên khắp thế giới. Sức mạnh của nó là có thể mô phỏng hoạt động của các hệ vi điều khiển mà không cần thêm phần mềm phụ trợ nào. Sau đó, phần mềm iSIS có thể xuất file sang ARES hoặc các phần mềm vẽ mạch in khác.

Trong lĩnh vực giáo dục, iSIS có ưu điểm là hình ảnh mạch điện đẹp, cho phép ta tùy chọn đường nét, màu sắc mạch điện, cũng như thiết kế theo các mạch mẫu (templates).

A screenshot of a computer

Description automatically generated with medium confidence

Hình .. Giao diện chính trên proteus 8

Các tính năng của Proteus

* Vẽ sơ đồ nguyên lý:

Bạn có thể dễ dàng vẽ được các sơ đồ mạch điện tử trên Proteus một cách dễ dàng và nhanh chóng. Bạn lấy linh kiện mong muốn từ thư viện của Proteus, sau đó kết nối các linh kiện lại với nhau để tạo ra một mạch điện tử hoàn chỉnh. Để biết cách vẽ một sơ đồ nguyên lý bằng Proteus, bạn xem bài hướng dẫn [tại đây](https://dientuviet.com/huong-dan-su-dung-proteus/" \t "_blank).

* Mô phỏng:

Khả năng ứng dụng chính của Proteus là mô phỏng, phân tích các kết quả từ các mạch nguyên lý. Proteus giúp người sử dụng có thể thấy trước được mạch thiết kế chạy đúng hay sai trước khi thi công mạch.

Các công cụ phục vụ cho việc phân tích mạch có độ chính xác khá cao như vôn kế đo điện áp, ampe kế đo dòng điện, máy đao động ký.

Đối với các bạn yêu thích điện tử thì đây là công cụ tuyệt vời. Nó giúp cho các bạn tự học, tự nghiên cứu và thiết kế thử các mạch điện tử và chạy mô phỏng để kiểm tra kết quả từ đó rút ra được những bài học hữu ích. Điều này sẽ giúp bạn tiết kiệm được thời gian và tiền bạc khi bạn không có điều kiện mà lại ham học và nghiên cứu.

* Thiết kế mạch in PCB:

Là tính năng dễ sử dụng trong Proteus. Bạn có thể tự tạo bản thiết kế hoặc bắt Proteus làm hộ bạn. Tự tạo bản thiết kế rất dễ dàng chỉ cần bạn đặt những chi tiết vào sơ đồ và vẽ đường mạch điện chạy qua. Đừng lo lắng về việc vi phạm bất kỳ quy tắc thiết kế nào bởi vì nó sẽ tự động phát hiện ra lỗi. Còn nếu muốn Proteus làm thay bạn thì chỉ cần đặt các chi tiết vào vị trí tương ứng rồi cho chạy tự động. Nó sẽ vẽ ra các cách đặt đường mạch và lựa bản tốt nhất. Và hiện nay còn có một tùy chỉnh nữa “Auto placer”, nó yêu cầu bạn xác lập kích thước bảng bằng cách vẽ hình dáng và kích cỡ bản mạch. Sau đó, nó tự động đặt các chi tiết vào trong khuôn. Sau đó, tất cả việc bạn phải làm là lập sơ đồ mạch.

Ngoài ra, trong quá trình thiết kế mạch in bạn cũng có thể xem hình 3D. Tính năng này rất hữu ích, nó cho phép bạn thiết ra những board mạch in đẹp theo mong muốn của mình.

Một số ưu điểm của Proteus là:

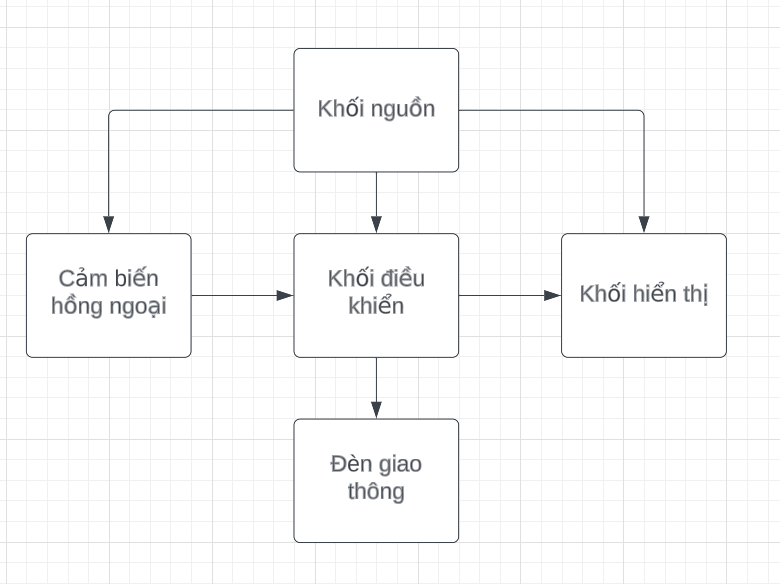
* Tự động sắp xếp đường mạch và vẽ điểm giao đường mạch.
* Chọn đối tượng và thiết lập thông số cho đối tượng dễ dàng.
* Xuất file thống kê linh kiện cho mạch.
* Xuất ra file Netlist tương thích với các chương trình làm mạch in thông dụng.
* Đối với người thiết kế mạch chuyên nghiệp, iSIS tích hợp nhiều công cụ giúp cho việc quản lý mạch điện lớn, mạch điện có thể lên đến hàng ngàn linh kiện.
* Thiết kế theo cấu trúc (hierachical design).
* Khả năng tự động đánh số linh kiện.

# Phân tích và thiết kế hệ thống

## Phân tích

***Sơ đồ khối***

Trên hình 3.1 là sơ đồ kết nối của hệ thống:

****

Hình .. Sơ đồ khối kết nối của hệ thống

**Chức năng của từng khối:**

* Khối nguồn:

Nguồn điện 1 chiều điện áp 5V cấp cho vi điều khiển, lcd hiển thị.

* Khối điều khiển trung tâm:

+ Vi điều khiển trung tâm

+ Phần mềm điều khiển.

* Khối hiển thị:

+ Dùng LCD nguồn 1 chiều để hiển thị lưu lượng xe qua các ngã tư.

+ Đèn phân làn, chỉ dẫn, đèn xang, đèn vàng, đèn đỏ

* Khối cảm biến (Cảm biến hồng ngoại):

Đóng vai trò là khối đầu vào: Dùng để đo lưu lượng xe và cung cấp thông tin dữ liệu truyền đến cho khối điều khiển.

* Đèn giao thông:

Hệ thống bộ 3 đèn led gồm 3 màu: xanh, vàng, đỏ dùng để hiển thị và báo hiệu cho người tham gia giao thông được phép đi và không được phép đi.

## Thuật toán giải quyết bài toán tắc đường

Bài toán: Giải quyết tắc đường cho ngã tư chính trong hệ thống giao thông cục bộ gồm 3 ngã tư.

Giả sử 3 ngã tư có vị trí lần lượt là *i* -1, *i* và *i* +1 và có các hướng W, S, E, N như hình vẽ dưới đây:

A picture containing diagram, screenshot, plan, rectangle

Description automatically generated

Hình .. Sơ đồ minh hoạ hệ thống

W

S

N

E

Ngã tư i+1

Ngã tư i-1

Ngã tư i

Hình .. Sơ đồ mô phỏng hệ thống

**Bài toán tắc đường:**

***Gọi:***

Chiều xe chạy từ W sang E là chiều WE

Chiều xe chạy từ E sang W là chiều EW

Chiều xe chạy từ S sang N là chiều SN

Chiều xe chạy từ N sang S là chiều NS

Giả sử các làn đường có diện tích như nhau.

Chiều WE và EW gồm có 3 làn đường; chiều SN và NS gồm có 2 làn dường.

Do xe chạy tại các ngã tư luôn luôn được chạy theo 2 hướng: rẽ phải, đi thẳng.

* Tại các ngã tư có 4 đèn giao thông luôn luôn tương ứng với trạng thái:

2 đèn đỏ - 2 đèn xanh.

Tính hệ số tắc đường = ( trong 1 phút)

**Mô tả thuật toán giải quyết tắc đường cho 1 hệ thống gồm 3 ngã tư liên tiếp nhau:**

Sử dụng vòng lặp liên tục:

Đầu tiên, khởi tạo biến thời gian mặc định cho các cột đèn giao thông. Sau đó, bật các đèn theo các màu đã được sắp xếp trước đó. Nhận thông tin từ các cảm biến hồng ngoại.

Tại các ngã tư, sử dụng cảm biến để đếm số lượng phương tiện trong vòng 1 phút. Kiểm tra lưu lượng xe tại các chiều lưu thông đó, xem có tắc hay không? Nếu xảy ra tắc đường tại 1 ngã tư thì: cần tính hệ số tắc đường tại các hướng tắc.

Sử dụng vòng lặp con:

Tìm xem hướng đường nào tắc hơn bằng cách tìm hệ số tắc đường lớn nhất. Sau đó, tiếp tục kiểm tra trạng thái đèn xem tại hướng tắc nhất đó: Nếu đang ở đèn xanh, thì tiếp tục tính lại xem sau *n* giây đèn xanh đó thì hệ số tắc đường là bao nhiêu. Sau đó quay lại, tìm lại hệ số tắc đường lớn nhất và tiếp tục vòng lặp. Nếu sau khi hết đèn xanh mà hệ số tắc đường ở hướng này vẫn lớn nhất thì tăng đèn xanh tại hướng này lên 10 giây, đồng thời tăng đèn đỏ ở tại 3 đèn còn lại lên 15 giây. Còn nếu ở hướng này đang là đèn đỏ thì cần kiểm tra xem đèn nào đang xanh. Sau đó, lập tức chuyển đèn đó sang trạng thái vàng để chuẩn bị chuyển sang đỏ. Rồi, bật đèn xanh cho đường đang tắc được giải phóng.

Nếu không xảy ra tắc đường thì các đèn hoạt động bình thường.

A diagram of a flowchart

Description automatically generated with low confidence

Hình .. Sơ đồ thuật toán giải quyết bài toán đặt ra

### Sơ đồ thuật toán giải quyết tắc đường:

**A picture containing diagram, sketch, technical drawing, plan

Description automatically generated**

Hình .. Thuật toán 1: Sơ đồ giải phóng làn đường bị tắc

Mô tả thuật toán 1: Thuật toán giải phóng làn đường bị tắc

Đầu tiên, nhận thông tin từ cảm biến hồng ngoại. Sau đó, tính các hệ số tắc đường tại ngã tư đó. Từ đó, tìm hệ số tắc đường lớn nhất. Sau khi tìm được hệ số tắc đường lớn nhất thì tiếp tục kiểm tra tín hiệu đèn tại làn đường đó. Nếu đang ở đèn xanh, thì tiếp tục tính lại xem sau *n* giây đèn xanh đó thì hệ số tắc đường là bao nhiêu. Sau đó quay lại, tìm lại hệ số tắc đường lớn nhất và tiếp tục vòng lặp. Nếu sau khi hết đèn xanh mà hệ số tắc đường ở hướng này vẫn lớn nhất thì tăng đèn xanh tại hướng này lên 10 giây, đồng thời tăng đèn đỏ ở tại 3 đèn còn lại lên 15 giây. Còn nếu ở hướng này đang là đèn đỏ thì cần kiểm tra xem đèn nào đang xanh. Sau đó, lập tức chuyển đèn đó sang trạng thái vàng để chuẩn bị chuyển sang đỏ. Rồi, bật đèn xanh cho đường đang tắc được giải phóng.

### Thuật toán đếm lưu lượng xe của cảm biến

Hình .. Thuật toán 2: Sơ đồ thuật toán đếm lưu lượng xe

Mô tả thuật toán 2: Thuật toán đếm lưu lượng xe trên đường:

Giả sử các xe trên đường đang di chuyển với tốc độ như nhau. Thuật toán được đo trong vòng khoảng 1phút. Đầu tiên chúng ta khời tạo và đọc giá trị mặc định. Sử dụng vòng lặp:

- Lưu giá trị biến đếm

- Khởi tạo biến dem=0;

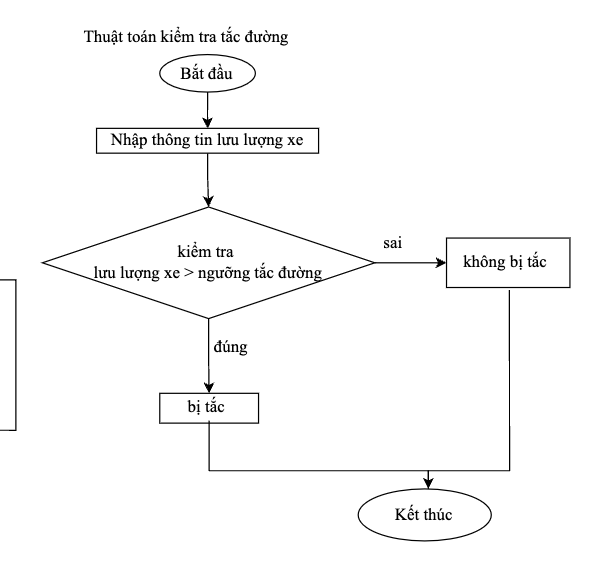
- Nếu

+ có xe đi qua mà cảm biến quét được thì biến dem:=dem+1;

+ không thì dem:=dem;

Sau đó, quay trở lại vòng lặp lưu giá trị biến đếm. Chúng ta sẽ liên tục đếm trong vòng 1 phút. Hết 1 phút thì sẽ lặp lại vòng lặp .Sau đó sẽ cho ra được kết quả lưu lượng xe trong vòng 1 phút .

### Thuật toán kiểm tra tắc đường



Hình .. Thuật toán 3: Sơ đồ thuật toán kiểm tra tắc đường

**Mô tả thuật toán 3: Thuật toán đếm lưu lượng xe trên đường:**

Đầu tiên, nhận thông tin lưu lượng xe mà các cảm biến đã đếm được.

Sau đó, kiểm tra xem: nếu lưu lượng xe > ngưỡng tắc đường:

Thì trả về giá trị 1, tức là có tắc đường

Không thì trả về giá trị 0, tức là không tắc đường.

### Thuật toán xác định làn đường bị tắc

Thuật toán 4: Thuật toán xác định làn đường bị tắc

Input: Lưu lượng xe của 12 tuyến T={T1,....,T12}

Output: Tắc hoặc không tắc

B1: Nhâp vào 1 mảng lane[12]

B2: For (i=0;i<length(lane)+1;i+1)

B3: If( Ti> Ti.threshold)

B4: then: lane[i] =1

B5. else lane[i]=0

B6: End

**Mô tả thuật toán 4: Thuật toán xác định làn đường bị tắc:**

Đầu tiên, nhập vào 1 mảng gồm 12 phần tử tương ứng với 12 chiều lưu thông tại 3 ngã tư .

+ Ngã tư *i*-1 gồm 4 lane: lane[0] có chiều từ W sang E lane[1] có chiều từ S sang N, lane[2] có chiều từ E sang W, lane[3] có chiều từ N sang S.

+ Ngã tư *i* gồm 4 lane: lane[4] có chiều từ W sang E lane[5] có chiều từ S sang N, lane[6] có chiều từ E sang W, lane[7] có chiều từ N sang S.

+ Ngã tư *i*+1 gồm 4 lane: lane[8] có chiều từ W sang E lane[9] có chiều từ S sang N, lane[10] có chiều từ E sang W, lane[11] có chiều từ N sang S.

Sử dụng vòng lặp for cho *i* chạy từ 0 đến 11. Kiểm tra xem nếu lưu lượng xe tại đó lớn hơn ngưỡng tắc đường thì trả về giá trị 1, nếu không thì trả về giá trị 0.

## Các bài toán tắc đường

### Kịch bản 1: Tắc nghẽn 1 chiều tại ngã tư

Ở bài toán này, giả sử tại ngã tư thứ *i* có 4 chiều là: W,S,E,N tương ứng với các tín hiệu đèn, đang bị tắc nghẽn tại chiều từ W sang E.

* Khi đó chúng ta có thể suy ra rằng chiều đi từ W sang E của ngã tư thứ *i*-1 cũng đang bị tắc. Lúc này bài toán của chúng ta là giải quyết tắc nghẽn ngã tư thứ *i* và *i*-1.

Để giải quyết được bài toán này, thì trước tiên chúng ta phải điều chỉnh được lưu lượng xe đến từ 2 ngã tư liên kết với nó.

* Đầu tiên phải xem xét lưu lượng xe đến từ các ngã tư thứ *i* và ngã tư thứ *i*- 1.
* Sau đó đèn giao thông sẽ được điều chỉnh sao cho tình trạng tắc đường được giải quyết.
* Lúc này làn đường từ chiều W->E của ngã tư thứ *i* và *i*-1 sẽ được ưu tiên giải phóng. Kiểm tra tín hiệu đèn tại làn này, nếu đèn đang còn n giây đèn xanh thì tính lưu lượng xe sau n giây đèn xanh đó. Kiểm tra lại xem đường còn tắc hay không. Nếu còn tắc sẽ tăng thêm 10s đèn xanh, đồng thời tăng thời gian đèn đỏ tại các đèn tín hiệu khác lên 15 giây. Còn nếu tín hiệu đèn này đang đỏ, thì cần kiểm tra tín hiệu đèn xanh tại ngã tư đó, và chuyển tín hiệu đó sang vàng, sau đó sang đỏ để cho tín hiệu đèn của làn đường tắc được bật xanh.
* Nếu có một trong những làn đường còn lại vượt ngưỡng lưu lượng giới hạn thì thuật toán sẽ tạm dừng họat động ở ngã tư đó.

### Kịch bản 2: Tắc nghẽn 2 chiều tại ngã tư

Ở bài toán này, giả sử tại ngã tư thứ *i* có 4 chiều là: W,S,E,N tương ứng với các tín hiệu đèn, đang bị tắc nghẽn tại chiều từ W sang E và từ E sang W.

* Khi đó chúng ta có thể suy ra rằng chiều đi từ W sang E và từ E sang W của các ngã tư thứ *i*-1 và *i*+1 cũng đang bị tắc.
* Lúc này bài toán của chúng ta là giải quyết tắc nghẽn ngã tư thứ *i*, *i*+1 và *i*-1.

Để giải quyết được bài toán này, thì trước tiên chúng ta phải điều chỉnh được lưu lượng xe đến từ 2 ngã tư liên kết với nó.

* Đầu tiên phải xem xét lưu lượng xe đến từ các ngã tư thứ *i* – 1 và ngã tư thứ *i* + 1.
* Nếu như các làn đường theo chiều từ S->N và N->S của cả ba ngã tư trên không xảy ra tắc nghẽn thì lúc này các đèn tín hiệu của cả ba ngã tư trên sẽ được ưu tiên cho các làn đi từ W qua E và ngược lại từ E qua W.
* So sánh hệ số tắc đường của 2 làn đường đó, bên nào có hệ số tắc đường lớn hơn sẽ ưu tiên được giải phóng trước.
* Kiểm tra tín hiệu đèn tại làn này, nếu đèn đang còn n giây đèn xanh thì tính lưu lượng xe sau n giây đèn xanh đó. Kiểm tra lại xem đường còn tắc hay không. Nếu còn tắc sẽ tăng thêm 10s đèn xanh, đồng thời tăng thời gian đèn đỏ tại các đèn tín hiệu khác lên 15 giây. Còn nếu tín hiệu đèn này đang đỏ, thì cần kiểm tra tín hiệu đèn xanh tại ngã tư đó, và chuyển tín hiệu đó sang vàng, sau đó sang đỏ để cho tín hiệu đèn của làn đường tắc được bật xanh. Liên tục tính lại hệ số tắc đường để so sánh hệ số tắc đường của 2 làn đó. Làn nào có hệ số tắc đường lớn hơn sẽ ưu tiên được giải phóng trước.
* Nếu có một trong những làn đường còn lại vượt ngưỡng lưu lượng giới hạn thì thuật toán sẽ tạm dừng họat động ở ngã tư đó.

### Kịch bản 3: Tắc nghẽn 3 chiều tại ngã tư

Ở bài toán này, giả sử tại ngã tư thứ *i* có 4 chiều là: W,S,E,N tương ứng với các tín hiệu đèn, đang bị tắc nghẽn tại chiều từ W sang E và từ E sang W và từ S sang N.

* Khi đó chúng ta có thể suy ra rằng chiều đi từ W sang E và từ E sang W của các ngã tư thứ *i*-1 và *i*+1 cũng đang bị tắc
* Lúc này bài toán của chúng ta là giải quyết tắc nghẽn ngã tư thứ i, i+1 và i-1.

Để giải quyết được bài toán này, thì trước tiên chúng ta phải điều chỉnh được lưu lượng xe đến từ 2 ngã tư liên kết với nó.

* Đầu tiên phải xem xét lưu lượng xe đến từ các ngã tư thứ *i* – 1 và ngã tư thứ *i* + 1.
* Nếu như các làn đường theo chiều từ S->N và N->S của cả ba ngã tư trên không xảy ra tắc nghẽn thì lúc này các đèn tín hiệu của cả ba ngã tư trên sẽ được ưu tiên cho các làn đi từ W qua E và ngược lại từ E qua W.
* So sánh hệ số tắc đường của 3 làn đường đó, bên nào có hệ số tắc đường lớn hơn sẽ ưu tiên được giải phóng trước.
* Kiểm tra tín hiệu đèn tại làn này, nếu đèn đang còn n giây đèn xanh thì tính lưu lượng xe sau n giây đèn xanh đó. Kiểm tra lại xem đường còn tắc hay không. Nếu còn tắc sẽ tăng thêm 10s đèn xanh, đồng thời tăng thời gian đèn đỏ tại các đèn tín hiệu khác lên 15 giây. Còn nếu tín hiệu đèn này đang đỏ, thì cần kiểm tra tín hiệu đèn xanh tại ngã tư đó, và chuyển tín hiệu đó sang vàng, sau đó sang đỏ để cho tín hiệu đèn của làn đường tắc được bật xanh. Liên tục tính lại hệ số tắc đường để so sánh hệ số tắc đường của 3 làn đó. Làn nào có hệ số tắc đường lớn hơn sẽ ưu tiên được giải phóng trước.
* Nếu có một trong những làn đường còn lại vượt ngưỡng lưu lượng giới hạn thì thuật toán sẽ tạm dừng họat động ở ngã tư đó.

### Kịch bản 4:Tắc nghẽn tại tất cả 4 chiều của ngã tư

Ở bài toán này, giả sử tại ngã tư thứ *i* có 4 chiều là: W,S,E,N tương ứng với các tín hiệu đèn, đang bị tắc nghẽn tại tất cả các chiều từ W sang E từ E sang W, S sang N và N sang S.

* Lúc này đèn tín hiệu tại ngã tư thứ i sẽ được được điều khiển theo mặc định và các tín hiệu đèn giao thông tại các ngã tư nhỏ sẽ được ưu tiên giải phóng các làn từ W qua E tại *i*+1 và từ E qua W tại *i*-1.
* So sánh hệ số tắc đường của 4 làn đường đó, bên nào có hệ số tắc đường lớn hơn sẽ ưu tiên được giải phóng trước.
* Kiểm tra tín hiệu đèn tại làn này, nếu đèn đang còn n giây đèn xanh thì tính lưu lượng xe sau n giây đèn xanh đó. Kiểm tra lại xem đường còn tắc hay không. Nếu còn tắc sẽ tăng thêm 10s đèn xanh, đồng thời tăng thời gian đèn đỏ tại các đèn tín hiệu khác lên 15 giây. Còn nếu tín hiệu đèn này đang đỏ, thì cần kiểm tra tín hiệu đèn xanh tại ngã tư đó, và chuyển tín hiệu đó sang vàng, sau đó sang đỏ để cho tín hiệu đèn của làn đường tắc được bật xanh. Liên tục tính lại hệ số tắc đường để so sánh hệ số tắc đường của 4 làn đó. Làn nào có hệ số tắc đường lớn hơn sẽ ưu tiên được giải phóng trước.
* Nếu có một trong những làn đường còn lại vượt ngưỡng lưu lượng giới hạn thì thuật toán sẽ tạm dừng họat động ở ngã tư đó.

## Nguyên lý hoạt động

Nguyên lý hoạt động của mạch như sau:

- Cảm biến hồng ngoại được sử dụng để phát hiện năng lượng phát ra từ xe cộ, mặt đường và các vật thể khác. Năng lượng được thu bởi các cảm biến hồng ngoại này được tập trung vào một vật liệu nhạy cảm với tia hồng ngoại bằng cách sử dụng một hệ thống quang học, sau đó chuyển đổi năng lượng thành tín hiệu điện.

- Các tín hiệu này được gắn dưới mặt đường để xem lưu lượng. Cảm biến hồng ngoại được sử dụng để điều khiển tín hiệu, phát hiện xe đi qua đường và truyền thông tin giao thông.

- Tại thời điểm ban đầu chưa có vật cản đi qua, cảm biến hồng ngoại không nhận được tín hiệu do đó không có tín hiệu đưa đến vi điều khiển. Màn hình LCD trong trường hợp này hiển thị giá trị đếm nhớ trước đó.

- Khi có vật cản đi qua, cảm biến hồng ngoại thu nhận được tín hiệu và truyền tín hiệu đến vi điều khiển Arduino, vi điều khiển thực hiện tác vụ tăng giá trị đếm và xuất hiển thị giá trị đếm lên mà hình LCD đồng thời đảm bảo khi một vật đứng yên tại vị trí của cảm biến hồng ngoại thì số đếm không tăng lên và hiển thị lên LCD.

- Nhược điểm cơ bản của cảm biến hồng ngoại là hoạt động của hệ thống có thể bị ảnh hưởng do sương mù; việc cài đặt và bảo trì hệ thống cũng tẻ nhạt.

## Thiết kế hệ thống trên Proteus

Hình 3-3. mô phỏng sơ đồ nối chân cụ thể giữa các khối của hệ thống trên thực tế bằng phần mềm Proteus 8 Professional.

A picture containing diagram, plan

Description automatically generated

Hình .. Mô phỏng mạch trên phần mềm Proteus

Giải thích nguyên lý hoạt động:

* Ở thời điểm ban đầu,cảm biến chưa có vật cản đi qua (TestPin =1) thì không có tín hiệu đưa đến vi điều khiển. Màn hình LCD trong trường hợp này hiển thị giá trị đếm nhớ trước đó,thường sẽ là 0.
* Khi có vật cản đi qua (TestPin=0) thì lập tức có tín hiệu từ cảm biến gửi đến Arduino rồi vi điều khiển thực hiện tác vụ tăng giá trị đếm và xuất,hiển thị giá trị đếm lên màn hình LCD.
* Như được mô tả trong hình 3.8, mỗi tín hiệu có thể chứa thông tin liên quan đến mọi phương tiện đi qua nó. Do đó, khi một phương tiện đi qua tín hiệu, tín hiệu có thể tự động giữ số lượng phương tiện đi qua nó và giúp phát hiện tắc nghẽn giao thông. Mỗi tín hiệu phải được lưu trữ với một giá trị ngưỡng có màu đỏ và xanh lục. Giờ đây, tùy thuộc vào tần suất của các phương tiện đi qua tín hiệu mỗi giây, bộ hẹn giờ có thể được điều khiển linh hoạt. Mỗi bộ điều khiển của tín hiệu phải được lưu trữ một giá trị tần số tối thiểu của các phương tiện đi qua tín hiệu. Ngay khi đạt đến tần số tối thiểu này, bộ điều khiển sẽ gửi lệnh đến tín hiệu để chuyển sang màu đỏ. Như vậy tín hiệu được điều khiển động.

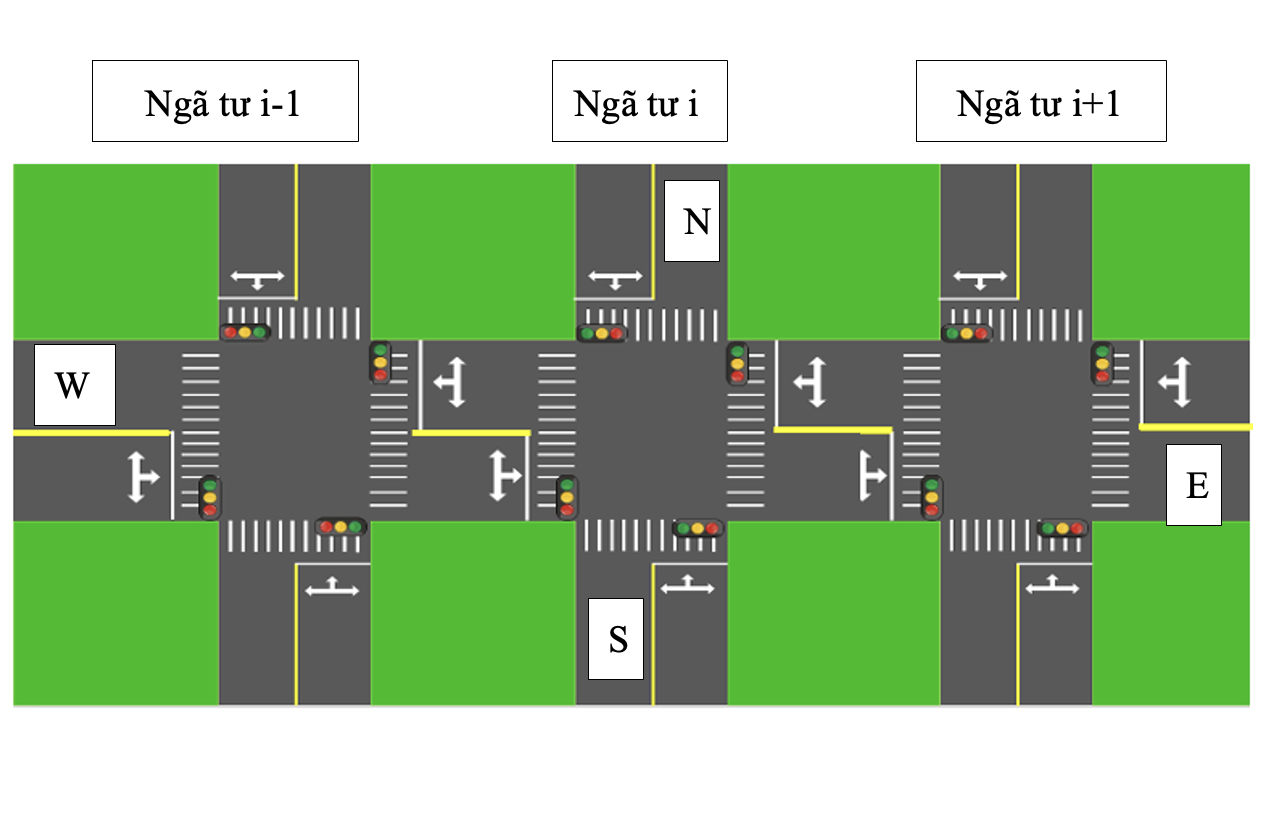
Ví dụ: giả sử đối với một tín hiệu, thời gian tối đa mà tín hiệu có thể có màu đỏ được đặt là 30 giây và thời gian tối đa mà tín hiệu có thể có màu xanh lục được đặt là 20 giây. Bộ điều khiển được lưu trữ với giá trị tần suất tối thiểu của các phương tiện đi qua nó mỗi giây là 5. Bây giờ, giả sử tín hiệu chuyển sang màu xanh lục, bộ hẹn giờ bắt đầu với giá trị tối đa là 20. Ban đầu tần số của các phương tiện đi qua tín hiệu mỗi giây là 10 , sau 10 giây, tần số này giảm xuống còn 5.

* Sau đó bộ điều khiển cảm biến sẽ tự động gửi lệnh đến tín hiệu để chuyển sang màu đỏ. Do đó, tín hiệu chuyển sang màu đỏ và tín hiệu liền kề của nó trong đường giao nhau đó chuyển sang màu xanh lục. Quá trình này tiếp tục trong một chu kỳ. Do đó, điều khiển động của tín hiệu giúp giảm lãng phí thời gian. Điều này cũng giúp tránh tắc nghẽn giao thông vì ưu tiên cho một con đường có lưu lượng xe cộ cao. Hệ thống này giúp phát hiện tắc nghẽn giao thông. Nếu tần suất xe chạy qua tín hiệu trên giây vẫn cao hơn giá trị đã đặt mặc dù đạt đến giá trị cực đại của bộ định thời thì tắc nghẽn đã xảy ra tại điểm đó. Sau khi phát hiện tắc nghẽn, bộ điều khiển cảm biến có thể gửi một tin nhắn đến bộ điều khiển của tín hiệu trước đó để thông báo rằng nó tạm thời dừng lưu lượng dọc theo đoạn đường đó. Sau khi nhận được thông báo từ tín hiệu kế tiếp, bộ điều khiển cảm biến sẽ BẬT tín hiệu màu đỏ cho đoạn đường hướng tới điểm giao nhau tắc nghẽn đó trong một khoảng thời gian xác định trước. Khi tình trạng tắc nghẽn được giải phóng tại ngã tư, bộ điều khiển của tín hiệu tương ứng sẽ gửi một thông báo khác đến bộ điều khiển trước đó của nó cho biết tiếp tục lưu lượng giao thông trở lại theo hướng đó. Chấp nhận thông báo này, bộ điều khiển của tín hiệu trước đó TẮT đèn đỏ và BẬT tín hiệu xanh và khởi động lại chu kỳ tín hiệu như trước.

# Triển khai và đánh giá kết quả

## Triển khai các thông số mô phỏng:

**Xây dựng bài toán:**

Thiết kế 1 hệ thống gồm 3 ngã tư liên tiếp nhau như hình vẽ duới đây:

Hình .. Sơ đồ minh hoạ hệ thống

Đông

Đông

Giả sử thời gian chờ đèn tại các ngã tư là như nhau:

* 30 giây đèn xanh
* 50 giây đèn đỏ
* 3 giây đèn vàng

Nếu có xảy ra tắc đường thì sẽ linh động thay đổi được giá trị của các đèn:

* xanh
* đỏ

còn lại thời gian chờ đèn vàng sẽ không đổi.

Giả sử lưu lượng xe trong 1 phút tại các mức khác nhau:

* Thấp : sẽ dao động từ khoảng 1 đến 15 xe.
* Trung bình: sẽ dao động từ khoảng 16 đến 25 xe .
* Cao: trên 25 xe.

Nếu lưu lượng xe vượt quá 30 xe => tắc đường.

Trung bình 10 xe qua ngã tư sẽ mất khoảng 30s .

**Các bài toán tắc đường**

**Kịch Bản 1: Tắc 1 chiều**

Giả sử lưu lượng tại chiều bị tắc đang là 45 xe và các làn đường còn lại trên ngã tư này trong khoảng từ 5 đến 15 xe. Lúc này hệ thống sẽ ghi nhận xảy ra ách tắc tại làn đường có lưu lượng xe vượt quá ngưỡng cho phép. Sau khi tiếp nhận thông tin hệ thống sẽ triển khai vòng lặp nhằm điểu khiển tín hiệu giao thông để giải quyết tình trạng tắc đường hiện thời đang xảy ra trên ngã tư này. Ngay khi các điều kiện để triển khai vòng lập là đúng thì trạng thái đèn giao thông trên ngã tư này sẽ được kiểm tra và điều chỉnh.

Mỗi 10 giây sẽ có 4 phương tiện được giải phóng tại các làn đường đang có tín hiệu màu xanh và 2 phương tiện sẽ nhập làn đường có tín hiệu màu đỏ. Do đó lưu lượng lưu thông của làn đường đang bị tắc này sẽ được biểu diễn như sau.

tắc này sẽ được biểu diễn như sau.

Chúng ta có thể thấy được sau 20 giây được cộng thêm tức là 50 giây cho đèn xanh thì tỉnh trạng tắc đường tại chiều này được giảm đi đáng kể và lưu lượng lưu thông liên tục ở dưới mức cảnh báo động và lưu lượng này sẽ còn được giảm dần đi theo thời gian. Trong khi đó nếu đèn giao thông được sắp xếp theo thời gian mặc định thì phải mất đến 190 giây để lưu lượng lưu thông ở chiều này đạt dưới mức báo động.

**Kịch Bản 2: Tắc 2 chiều**

Giả sử lưu lượng tại chiều bị tắc tại ngã tư *i* đang là 47 xe, lưu lượng tại chiều cùng hướng với chiều bị tắc tại ngã tư *i*-1 đang là 45 và các chiều còn lại trên 2 ngã tư này trong khoảng từ 5 đến 15 xe. Lúc này thuật toán sẽ được triển khai tại cả 2 ngã tư *i* và *i*-1.

Giống như kịch bản trên mỗi 10 giây sẽ có 4 phương tiện được giải phóng tại các chiều đang có tín hiệu màu xanh và 2 phương tiện sẽ nhập các làn đường có tín hiệu màu đỏ. Do đó lưu lượng lưu thông của chiều đang bị tắc này sẽ được biểu diễn như sau.

Chúng ta có thể thấy được sau 20 giây được cộng thêm tức là 50 giây cho đèn xanh chiều bị tắc tại ngã tư *i*-1 và thêm 30 giây tức 60 cho đèn xanh tại ngã tư *i* thì tỉnh trạng tắc đường tại 2 chiều này được giảm đi đáng kể và lưu lượng lưu thông liên tục ở dưới mức cảnh báo động và lưu lượng này sẽ còn được giảm dần đi theo thời gian. Trong khi đó nếu đèn giao thông được sắp xếp theo thời gian mặc định thì phải mất đến 190 giây để lưu lượng lưu thông tại chiều đang bị tắc tại *i-1* đạt dưới mức báo động và theo tính toán để lưu lượng lưu thông tại chiều bị tắc tại ngã tư *i* đạt dưới mức báo động sẽ phải mất tới 270 giây.