

LỜI MỞ ĐẦU

Hiện nay số lượng các phương tiện tham gia giao thông tại Việt Nam ngày một tăng, nhất là ở những thành phố lớn như Hà Nội và TP Hồ Chí Minh. Trong những năm gần đây cùng với sự phát triển của nền kinh tế là tốc độ gia tăng không ngừng về các loại phương tiện giao thông. Sự phát triển nhanh chóng của các phương tiện đã dẫn đến tình trạng tắc nghẽn giao thông hay xảy ra thường xuyên vào nhiều thời điểm. Vấn đề đặt ra ở đây là làm sao để đảm bảo tình trạng giao thông được thông suốt và sử dụng đèn giao thông tại các ngã tư, những nơi giao nhau của các làn đường là một giải pháp bức thiết rất cần được giải quyết. Lượng xe trên các tuyến đường có lưu lượng cao sẽ tích lũy theo thời gian, là một trong những nguyên nhân cơ bản dẫn đến tắc nghẽn. Điều này không chỉ gây lãng phí về thời gian, nhiên liệu mà còn ảnh hưởng xấu đến sức khỏe, tâm lý người dân và môi trường sinh thái.

Mục đích của mô hình là điều khiển hệ thống giao thông giúp cho các nút giao thông không bị ùn tắc, phân luồng cho các xe chạy theo hàng lối tránh xảy ra tai nạn đáng tiếc khi tham gia giao thông.

Dựa trên phương pháp nghiên cứu và phân tích đặc tính chức năng của các linh kiện, các IC và áp dụng những kiến thức đã học cùng với sự hướng dẫn của giảng viên phụ trách để xây dựng nên một mô hình điều khiển đèn tín hiệu giao thông thông minh, quản lý hoạt động tốt và đúng với yêu cầu của đề tài.

CHƯƠNG 1.

TỔNG QUAN VỀ MÔ HÌNH THIẾT KẾ

1.1. TÍNH CẤP THIẾT CỦA ĐỀ TÀI

Ngày nay, cảnh quan đô thị Hà Nội được nhớ tới bởi sự gia tăng đáng kể các phương tiện giao thông cá nhân. Xe hơi và xe máy đang làm các đường phố dường như nhỏ hẹp hơn và khiến hoạt động giao thông trở nên khó khăn hơn. Để giảm ùn tắc, chính phủ Việt Nam đã qui hoạch một Hệ thống giao thông vận tải công cộng (GTCC), bao gồm các tuyến tàu điện ngầm và những tuyến đường dành riêng cho xe buýt. Tuy nhiên, vấn nạn tắc nghẽn giao thông và ô nhiễm môi trường đang ngày càng trở nên nghiêm trọng hơn. Trong khi đó, mạng lưới vận tải công cộng, hiện đang trong quá trình chuẩn bị triển khai, lại có tiến độ rất chậm.

Hồi giữa những năm 1990, các học giả miêu tả Hà Nội là thành phố nơi hoạt động giao thông đang chuyển từ xe đạp sang xe máy (Godard, 1996). Hồi giữa những năm 2000, các nhà nghiên cứu nhận định Hà Nội là một “thành phố phụ thuộc vào xe máy) và vấn đề tắc nghẽn giao thông đang ngày càng trở nên nghiêm trọng hơn bao giờ hết. Thủ đô của Việt Nam, vốn được coi là “thành phố phát triển nhanh của thế giới” đang phải đối mặt với vấn nạn tắc nghẽn giao thông trên diện rộng giống như nhiều thành phố khác ở châu Á, đặc biệt là những thành phố đã từng trải qua việc xe hơi hóa nhanh chóng (Barter, 2000). Liên quan tới vấn đề này, rất nhiều giải pháp đối phó đã được chính quyền thành phố áp dụng như: nâng cấp mạng lưới đường bộ hiện có, xây dựng mới các con đường, cầu vượt và hầm ngầm; tổ chức lại giao thông ở các điểm giao cắt; thực thi luật giao thông; lắp đặt hệ thống đèn tín hiệu giao thông; cải thiện hoạt động của hệ thống giao thông công cộng hiện nay. Nhưng, dù đã có những khoản đầu tư đó, tình hình giao thông vẫn không

được cải thiện là bao. Cách đây một vài năm các cấp chính quyền đã có một số quyết định nhằm giảm sự chuyên đổi nhanh chóng phương tiện giao thông, Việc xây dựng một Hệ thống giao thông công cộng (dựa trên mạng lưới đường sắt có sức chuyên chở lớn/ đường sắt nhẹ (MRT/MRT) và các tuyến xe buýt tốc độ cao) dường như là câu trả lời thỏa đáng nhằm đạt được mục tiêu một hệ thống giao thông đô thị bền vững. Nhưng đã xuất hiện những hoài nghi vì tốc độ triển khai quá chậm và vì những khó khăn mà thành phố phải giải quyết nhằm hoàn tất việc xây dựng một Hệ thống vận tải công cộng liên kết.

Sau khi đề tài được đưa vào ứng dụng trong thực tế sẽ giải quyết được vấn đề ùn tắc giao thông ở nhiều nơi.

1.2. HỆ THỐNG ĐÈN GIAO THÔNG THÔNG MINH TRÊN THẾ GIỚI

Trong những năm gần đây, thế giới nói nhiều đến sự cần thiết phải có một Hệ thống giao thông thông minh (Intelligent Transport System - ITS). Về thực chất, ITS là ứng dụng công nghệ cao điện tử, tin học và viễn thông để điều hành và quản lý hệ thống giao thông vận tải. Tại một số nước phát triển, tự động hoá truyền tin trong giao thông vận tải đã được triển khai hàng chục năm nay.

Mục tiêu của ITS là gì? ITS được coi là một hệ thống lớn, trong đó con người, phương tiện giao thông, mạng lưới đường giao thông là các thành phần của hệ thống, liên kết chặt chẽ với nhau nhằm bảo đảm cho hệ thống giao thông vận tải đạt các mục tiêu: giảm tai nạn giao thông, giảm ùn tắc giao thông, giảm ô nhiễm môi trường, hạ giá thành vận chuyển; tăng hiệu quả vận chuyển, tạo điều kiện thuận lợi tối đa cho việc đi lại... Nhật Bản là một điển hình thực hiện ITS.

Bước khởi đầu để triển khai ITS tại Nhật Bản, các thông tin về giao thông được cung cấp qua Hệ thống thông tin liên lạc một phương tiện giao

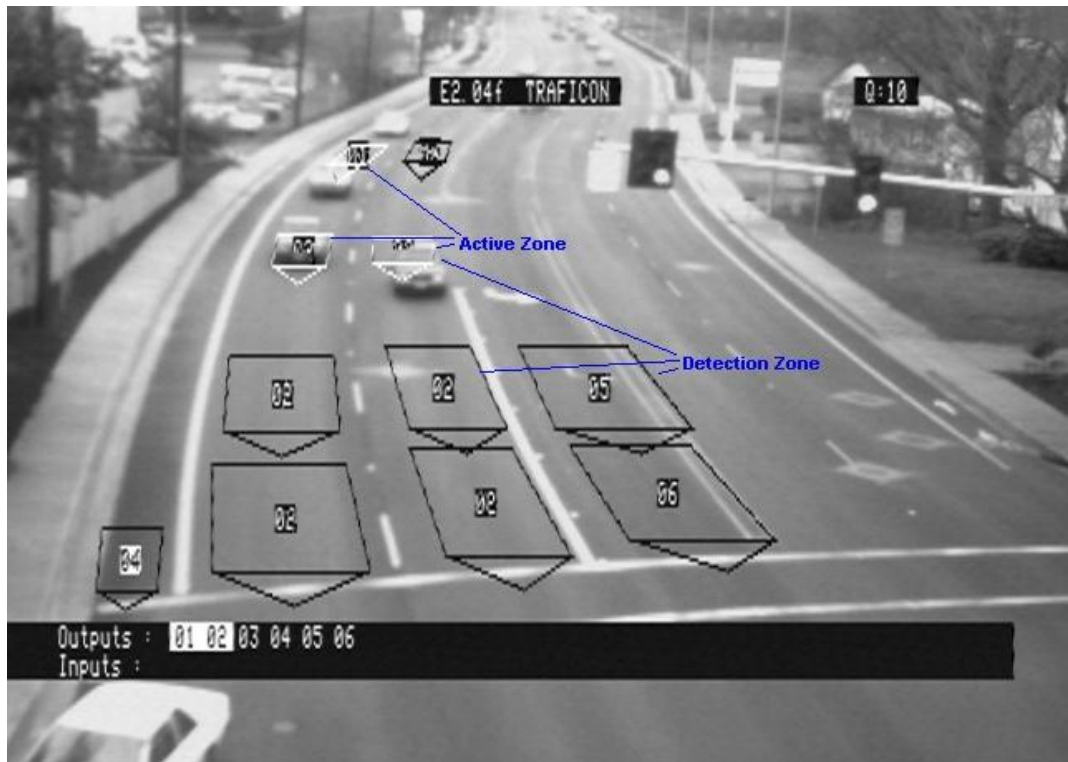
thông (VICS). Đây là một hệ thống dữ liệu số nhằm cung cấp cho các lái xe thông tin cập nhật về giao thông đường bộ. Sử dụng hệ thống này, thông tin chi tiết về đường bộ cần thiết cho lái xe được truyền đi từ cột tín hiệu đặt trên đường tới hệ thống thiết bị định vị đặt trên xe. Thông tin truyền đi trên diện rộng được thông qua đài phát sóng FM. Từ 1996-1998, số lượng hệ thống VICS bán ra đã đến 600.000 chiếc. Nơi được trang bị đầu tiên là các đường phố của thủ đô Tokyo. Hệ thống hỗ trợ lái xe tự động trên đường cao tốc (AHS) để được nghiên cứu và phát triển từ năm 1991. Mục tiêu nghiên cứu là cảnh báo những nguy hiểm phía trước trên đường, xác định vị trí của các phương tiện giao thông khác, ngăn ngừa va đập đằng sau. AHS được nghiên cứu trên 3 lĩnh vực chủ yếu: Thông tin: nghiên cứu việc cung cấp thông tin cho lái xe; Điều khiển: nghiên cứu hỗ trợ điều khiển xe; Dẫn đường tự động: nghiên cứu hỗ trợ lái xe hoàn toàn tự động. Sự an toàn của lái xe là trách nhiệm của hệ thống này. Dự án Phương tiện giao thông an toàn cao (ASV) cũng đã bắt đầu được nghiên cứu từ 1991 bao gồm 6 lĩnh vực và 32 hệ thống. Nhiều kết quả đã đạt được trong phát triển công nghệ tự động. Một số nhà sản xuất ô tô đã bán ra các hệ thống điều khiển dẫn đường thích ứng. Hệ thống thu thuế đường điện tử để chống ùn tắc giao thông (ETS) đã được nghiên cứu từ 1990 và triển khai từ tháng 3-1997. Hệ thống này của Nhật Bản phù hợp với tất cả các kiểu thu thuế đường trong khi sử dụng cùng một thiết bị trên xe. Giai đoạn từ 2000 đến nay thực sự là một cuộc cách mạng trong hệ thống giao thông với các dịch vụ của ITS cho người sử dụng ở Nhật. Từ năm 2005-2010, theo chương trình đã xây dựng, ITS sẽ kết hợp công nghệ mới với nâng cấp cơ sở hạ tầng, hoàn thiện pháp chế và thể chế xã hội, lái xe tự động sẽ trở thành hiện thực, hạn chế tối đa ùn tắc giao thông. Nhật Bản đất hẹp người đông nhưng rất tự hào vì đã áp dụng công nghệ phát triển các dữ liệu điều khiển bằng máy tính cho hàng triệu ô tô, tạo nên một phương tiện đi lại thông minh nhất thế giới.

Thông minh hóa hệ thống giao thông chính là giảm vai trò của con người trong điều hành giao thông. Khi con người không còn vai trò gì thì sẽ đạt đến mức tự động hóa. Đây chính là mục tiêu cao nhất của ITS. Để đạt được mục tiêu này, ITS phải có 3 giai đoạn: thu thập thông tin, xử lý thông tin và đưa thông tin được xử lý tới người tham gia giao thông.



Hình 1.1. Hệ thống ITS Hong Kong sử dụng hệ thống CCTV và các cảm biến để thu thập thông tin.

Chính vì vậy, sẽ có nhiều camera và cảm biến được lắp đặt trên mặt đường để thu thập các thông tin về luồng giao thông, khí hậu, thời tiết,... các thông tin này được hệ thống máy tính phân tích và xử lý, sau đó cung cấp trở lại cho tài xế về tình hình giao thông trên đường để tài xế chọn giải pháp giao thông tối ưu, giúp hạn chế tối đa tai nạn và ùn tắc giao thông, đảm bảo thời gian đi lại ngắn nhất và an toàn nhất cho các phương tiện đang lưu thông trên đường. Đây là hệ thống bao gồm các phương tiện truyền hình, nối mạng quản lý toàn quốc. Nhà quản lý chỉ cần ngồi một chỗ vẫn có thể bao quát được toàn bộ hệ thống đường toàn quốc. Đơn cử, một sự kiện đang xảy ra trên một điểm của đường cao tốc có thể lập tức được thông báo trong toàn hệ thống quản lý và sử dụng đường cao tốc, đồng thời kết nối với tổ chức thanh tra giao thông trên toàn quốc để kịp thời xử lý.



Hình 1.2: Hệ thống camera thông minh còn nhận biết được làn đường



Hình 1.3: Một trung tâm giám sát giao thông tại Nhật Bản

Tại Nhật, thậm chí người đi bộ cũng dùng máy định vị từ xa. Trong các dự án nghiên cứu gần đây, công nghệ giọng nói kết hợp mắt kính và tai nghe cho người mù để thu nhận các tín hiệu tia hồng ngoại. Khi nhận được các tín hiệu giao thông, hệ thống sẽ phát ra âm thanh "đỏ, đỏ, đỏ" hoặc "xanh, xanh

xanh" để cảnh báo người sử dụng khi qua đường. Nhật Bản không phải là quốc gia duy nhất áp dụng hệ thống vận tải thông minh trên đường phố. Tại Mỹ, cơ quan nghiên cứu ITS đã được thành lập từ năm 1990 với tên gọi "Hiệp hội phương tiện giao thông thông minh đường bộ Mỹ". Người lái xe Mỹ sẽ được giảm một số lệ phí nếu thanh toán bằng điện tử. Từ năm 1992, Liên minh châu Âu (EU) đã có nhiều chương trình, dự án nghiên cứu, ứng dụng công nghệ thông tin trong giao thông vận tải. Năm 1998, Hàn Quốc là nước chủ nhà tổ chức Hội nghị quốc tế về ITS và công bố chương trình tổng thể quốc gia về ITS do – Bộ Giao thông vận tải và Xây dựng chủ trì. Năm 1996, Bộ Giao thông vận tải Trung Quốc đã thành lập Trung tâm nghiên cứu về ITS và hệ thống quản lý giao thông tự động đã được triển khai ở Bắc Kinh. Malaixia đang đẩy mạnh phát triển ITS và đã có dự án tổng thể về ITS. Hệ thống ITS đã được lắp đặt ở một số tuyến đường thu thuế. Năm 1996, các cơ quan của Chính phủ Nhật Bản đã phối hợp soạn thảo "Chương trình tổng thể về ITS của Nhật Bản". Tại Xingapo, Chính phủ buộc người lái xe phải sử dụng máy thanh toán lệ phí cầu đường số.



Hình 1.4: Lắp đặt các Camera giám sát tại các nút giao thông

Tại Việt Nam, việc nghiên cứu và ứng dụng ITS là cần thiết, và phải làm từng bước cho phù hợp với điều kiện Việt Nam, để đến năm 2020 có cơ sở hạ tầng cho ITS như một số nước trong khu vực hiện nay.

1.3. Ý TƯỞNG THIẾT KẾ MÔ HÌNH ĐIỀU KHIỂN TÍN HIỆU ĐÈN GIAO THÔNG.

Dựa trên là một số hệ thống đèn giao thông thông minh hiện đại trên thế giới về tính năng tự động hóa của nó, qua đó chúng ta có thể tham khảo và ứng dụng vào trong thực tế. Tuy nhiên với thời gian, kiến thức và kinh phí hạn hẹp việc thiết kế một mô hình điều khiển giao thông thông minh còn khó khăn vì vậy em sẽ thiết kế một mô hình Đèn giao thông thu nhỏ phù hợp với thiết kế theo dạng điển hình , phổ thông hiện nay đang sử dụng tại nước ta. Ngoài ra đèn giao thông sẽ có thêm khả năng có thể thay đổi được thời gian các tuyến tùy vào lưu lượng người mà ta có thể thiết lập thời gian cho các tuyến tùy vào từng thời điểm. Thiết kế này rất phù hợp cho những tuyến đường, các điểm giao nhau vào giờ cao điểm, tan tầm,...và phần nào khắc phục được tình trạng ách tắc giao thông đang diễn ra hằng ngày tại các thành phố lớn trên cả nước.

1.4. CÁC PHƯƠNG ÁN THIẾT KẾ MẠCH ĐIỀU KHIỂN

1.4.1. Với mạch dùng IC rời

Ưu điểm:

- Đảm bảo độ chính xác cao
- Tần số đáp ứng của mạch nhanh
- Tổn hao công suất bé, mạch có thể sử dụng pin hoặc acquy
- Khả năng đếm rộng
- Giá thành thấp
- Mạch đơn giản dễ thực hiện

Nhược điểm:

Với việc sử dụng kỹ thuật số khó có thể đáp ứng được việc thay đổi số đếm. Muốn thay đổi một yêu cầu nào đó của mạch thì buộc lòng phải thay đổi phần cứng. Do đó mỗi lần phải lắp lại mạch dẫn đến tốn kém về kinh tế mà nhiều khi yêu cầu đó không thực hiện được bằng phương pháp này vì việc tính toán các thuật toán rất phức tạp.

1.4.2. Với mạch dùng PLC

Ưu điểm:

- Tần số đáp ứng của mạch nhanh
- Có khả năng khử nhiễu từ bên ngoài đảm bảo tín hiệu điều khiển
- Khả năng đếm rộng
- Đảm bảo độ chính xác cao
- Có thể thay đổi một cách linh hoạt bằng việc thay đổi phần mềm, trong khi đó phần cứng không cần thay đổi.

Nhược điểm:

- Vốn đầu tư khá cao
- Không phù hợp với làm mô hình nhỏ gây lãng phí về tài chính cũng như tài nguyên của PLC.

1.4.3. Với mạch kết nối với máy tính

Ưu điểm:

- Tốc độ xử lý rất lớn
- Đảm bảo độ chính xác cao
- Có thể thay đổi một cách linh hoạt bằng việc thay đổi phần mềm, trong khi đó phần cứng không cần thay đổi.

- Có thể quản lý từ xa qua mạng

Nhược điểm:

- Vốn đầu tư cao
- Chương trình điều khiển lập trình phức tạp.

- Lãng phí tài nguyên của máy tính vì chương trình sử dụng trong mô hình không phải xử lý những thuật toán quá phức tạp.

1.4.2. Với mạch dùng kỹ thuật vi điều khiển

Ngoài những ưu điểm như đã liệt kê trong phương pháp dùng IC rời thì mạch đếm sản phẩm dùng kỹ thuật vi điều khiển còn có những ưu điểm sau:

- Mạch có thể thay đổi số đếm một cách linh hoạt bằng việc thay đổi phần mềm, trong khi đó phần cứng không cần thay đổi mà mạch dùng IC rời không thể thực hiện được.

- Số linh kiện sử dụng trong mạch ít hơn nên giá thành thấp hơn.

- Mạch đơn giản hơn so với mạch dùng IC rời, PLC hay máy tính nhưng vẫn đảm bảo được tính năng điều khiển của mô hình.

⇒ Trong thiết kế người ta thường chọn phương pháp tối ưu và kinh tế do đó em chọn phương pháp xây dựng vi mạch dùng kỹ thuật vi xử lí.

CHƯƠNG 2.

NGHIÊN CỨU CÁC THIẾT BỊ SỬ DỤNG TRONG MÔ HÌNH

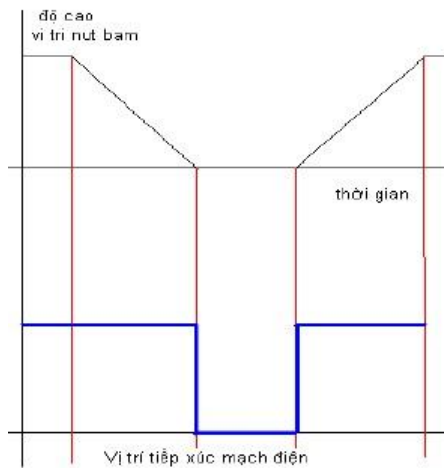
2.1. KHỐI PHÍM ĐIỀU KHIỂN BẰNG TAY

2.1.1. Công dụng của nút nhấn

Nút bấm là một hình thức ra lệnh phổ biến nhất hiện nay. Khi gọi một cái thang máy ta chỉ cần bấm nút, khi gọi cửa thì bấm chuông, bật đèn thì bấm nút công tắc, và điều khiển thời gian cho đèn tín hiệu giao thông cũng cần phải có nút nhấn để xuất lệnh điều khiển cho VXL . Lệnh rất đa dạng., bạn có thể bấm cùng một nút, nhưng lệnh sẽ khác nhau ở mỗi thời điểm, và mỗi trạng thái. Ví dụ, như bạn nhấp chuột máy tính, thực ra cũng là bạn nhấp nút bấm, nhưng bạn thấy rõ ràng rằng, ở những vị trí di chuyển chuột khác nhau, nút bấm của chuột sẽ đưa ra các mệnh lệnh khác nhau cho máy tính thực hiện.

2.1.2. Một số trạng thái bấm nút thông dụng

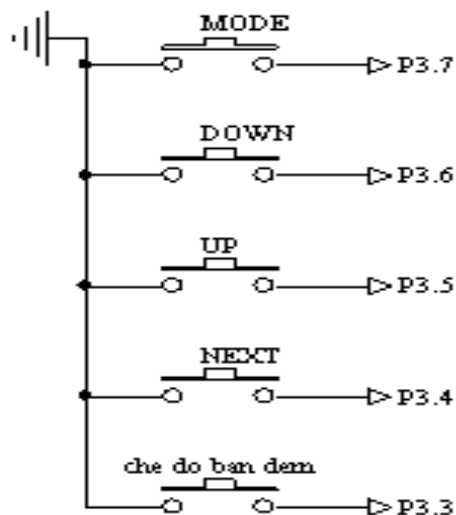
Trạng thái nút bấm ra lệnh tức thời, đó là khi ta bấm nút thì ngay lập tức mọi trạng thái phải được kiểm tra và chương trình dừng lại để thực hiện lệnh từ nút bấm. Có nghĩa là khi ta ra lệnh tại thời điểm bấm nút, và máy hiểu rằng đã bấm nút. Trạng thái chờ nút bấm, đó là chương trình mà ta đang chạy, đến một giai đoạn nào đó nó cần phải có sự ra lệnh bằng nút bấm, và chương trình chờ ra lệnh bấm nút để chạy tiếp, hoặc bắt đầu một công việc nào đó sau khi chờ. Khi bấm xuống nút nhấn sẽ có một giai đoạn đi xuống và khi chạm vào mạch điện phải có một khoảng thời gian giữ cho nút bấm tiếp xúc với mạch điện, sau đó là giai đoạn thả nút bấm ra.



Hình 2.1: Đường đặc tính của nút nhấn

Nhìn lên hình trên, ta thấy. Khi bấm nút, có quá trình đi xuống của nút bấm và quá trình đi lên của nút bấm. Nhưng thực tế, đối với mạch điện trong nút bấm, nó chỉ có thể nhận được trạng thái tiếp xúc hoặc không tiếp xúc, nên tín hiệu nhận được sẽ như đường màu xanh trong hình dưới. Chúng ta chỉ quan tâm đến trạng thái của đường màu xanh trong các ứng dụng của nút bấm. Vậy trạng thái nút bấm lại có thêm 3 trạng thái nữa là trạng thái bấm xuống, trạng thái giữ nút bấm và trạng thái nhả nút bấm lên. Kết hợp với 2 trạng thái điều khiển trên, chúng ta có 6 trạng thái phổ biến của nút bấm. Chúng ta lưu ý rằng, nút nhấn có 6 trạng thái chứ không phải chỉ có 4 trạng thái. Nhưng ở đây chúng ta xét trạng thái từ khi ta bấm nút cho tới khi nhả thì VĐK mới thực thi lệnh.

Trong mô hình này, ta có nút bấm được nối với một chân của VXL và chân còn lại được nối với GND. Những phím nhấn này được dùng để điều chỉnh thay đổi thời gian của đèn tín hiệu giao thông.



Hình 2.2: Sơ đồ khối nút nhấn đưa vào chân VDK

2.2. KHỐI XỬ LÝ TRUNG TÂM

2.2.1. Giới thiệu sơ lược về khối xử lý trung tâm

Bộ vi điều khiển viết tắt là Micro-controller, là mạch tích hợp trên 1 chip có thể lập trình được, dùng để điều khiển hoạt động của 1 hệ thống. Theo các tập lệnh của người lập trình, bộ vi điều khiển tiến hành đọc, lưu trữ thông tin, xử lý thông tin, đo thời gian và tiến hành đóng mở một cơ cấu nào đó.

Trong các thiết bị điện và điện tử dân dụng, các bộ vi điều khiển điều khiển hoạt động của tivi, máy giặt, điện thoại ... Trong hệ thống sản xuất tự động, bộ vi điều khiển được sử dụng trong Robot dây truyền tự động. Các hệ thống càng “thông minh” thì vai trò của hệ vi điều khiển càng quan trọng.

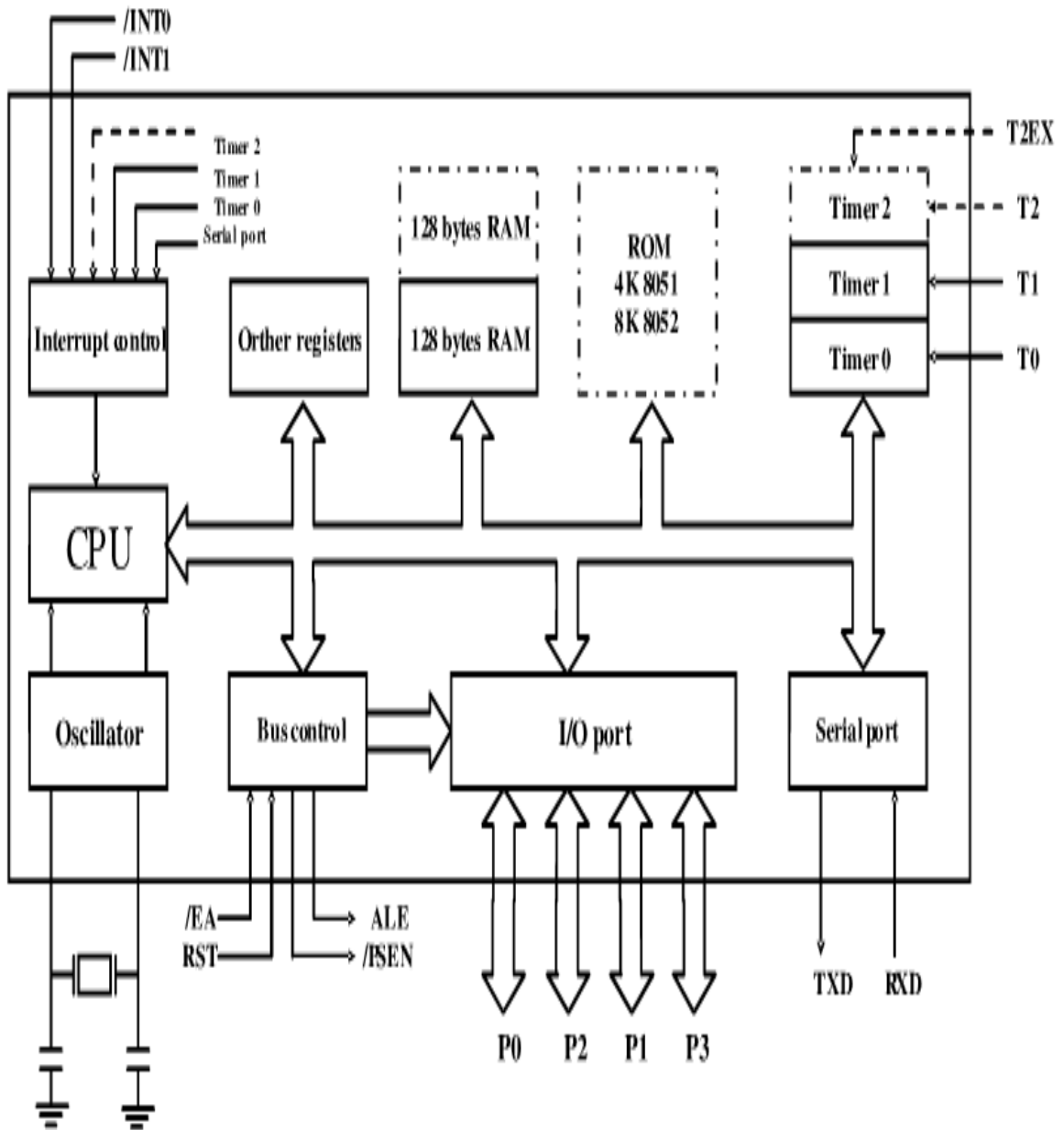
Với khối xử lý trung tâm này chúng em sử dụng IC vi điều khiển 89C52 là loại vi điều khiển thông dụng và chúng em đã được học tại trường.

2.2.2. Khảo sát bộ vi điều khiển 89C52

IC vi điều khiển 89C52 thuộc họ MCS51 có các đặc điểm sau:

- 8Kb ROM
- 256 byte RAM
- 4port I/O 8 bit
- 3 bộ định thời

- Giao tiếp nối tiếp
- 64Kb không gian bộ nhớ chương trình mở rộng
- 64Kb không gian bộ nhớ dữ liệu mở rộng



Hình 2.3: Cấu trúc phần cứng họ VDK 89C51

Chức năng các chân vi điều khiển:

(T2)	P1.0	□	1	40	□	VCC
(T2 EX)	P1.1	□	2	39	□	P0.0 (AD0)
	P1.2	□	3	38	□	P0.1 (AD1)
	P1.3	□	4	37	□	P0.2 (AD2)
	P1.4	□	5	36	□	P0.3 (AD3)
	P1.5	□	6	35	□	P0.4 (AD4)
	P1.6	□	7	34	□	P0.5 (AD5)
	P1.7	□	8	33	□	P0.6 (AD6)
	RST	□	9	32	□	P0.7 (AD7)
(RXD)	P3.0	□	10	31	□	\overline{EA}/VPP
(TXD)	P3.1	□	11	30	□	ALE/PROG
($\overline{INT0}$)	P3.2	□	12	29	□	\overline{PSEN}
($\overline{INT1}$)	P3.3	□	13	28	□	P2.7 (A15)
(T0)	P3.4	□	14	27	□	P2.6 (A14)
(T1)	P3.5	□	15	26	□	P2.5 (A13)
(\overline{WR})	P3.6	□	16	25	□	P2.4 (A12)
(RD)	P3.7	□	17	24	□	P2.3 (A11)
	XTAL2	□	18	23	□	P2.2 (A10)
	XTAL1	□	19	22	□	P2.1 (A9)
	GND	□	20	21	□	P2.0 (A8)

Hình 2.4: Sơ đồ chân cắm VDK 89C52

- Port 0:

Là port có chân từ 32 đến 39 có 2 công dụng. Trong các thiết kế có tối thiểu thành phần, port 0 được sử dụng làm nhiệm vụ xuất nhập. Trong các thiết kế lớn hơn có bộ nhớ ngoài, port 0 trở thành bus địa chỉ và bus dữ liệu đa hợp.

- Port 1:

Là các port có chân từ 1 đến 8. Có chức năng như các đường I/O.

- Port 2:

Là port có chân từ 21 đến 28 có 2 công dụng, hoặc làm nhiệm vụ xuất nhập hoặc là byte địa chỉ cao của bus địa chỉ 16 – bit cho các thiết kế có bộ

nhớ chương trình ngoài hoặc các thiết kế có nhiều hơn 256 byte bộ nhớ dữ liệu ngoài.

- Port 3:

Là các port có chân từ 10 đến 17. Có chức năng như các đường I/O. Ngoài ra còn có chức năng đặc biệt sau:

Bảng 2.1: Chức năng đặc biệt các chân Port 3

Bit	Tên	Địa chỉ bit	Chức năng
P3.0	RxD	B0H	Chân nhận dữ liệu của port nối tiếp
P3.1	TxD	B1H	Chân phát dữ liệu của port nối tiếp
P3.2	$\overline{INT0}$	B2H	Ngõ vào ngắt ngoài 0
P3.3	$\overline{INT1}$	B3H	Ngõ vào ngắt ngoài 1
P3.4	T0	B4H	Ngõ vào của bộ định thời / đếm 0
P3.5	T1	B5H	Ngõ vào của bộ định thời / đếm 1
P3.6	\overline{WR}	B6H	Điều khiển ghi bộ nhớ dữ liệu ngoài
P3.7	\overline{RD}	B7H	Điều khiển đọc bộ nhớ dữ liệu ngoài

- \overline{PSEN} (Program Store Enable):

Chân 29. Chân cho phép đọc bộ nhớ chương trình ngoài.

- ALE (Address Latch Enable):

Chân 30. Là chân tín hiệu cho phép chốt địa chỉ để truy cập bộ nhớ ngoài, khi On-chip xuất ra byte thấp của địa chỉ. Tín hiệu chốt được kích hoạt ở mức cao, tần số xung chốt = 1/6 tần số dao động của bộ VĐK. Nó có thể được dùng cho các bộ Timer ngoài hoặc cho mục đích tạo xung Clock. Đây cũng là chân nhận xung vào để nạp chương trình cho Flash (hoặc EEPROM) bên trong On-chip khi nó ở mức thấp.

- \overline{EA} (External Access):

Chân 31. Tích cực mức thấp, chạy chương trình ROM ngoài. Tích cực mức cao, chạy chương trình ROM nội.

- Các ngõ vào bộ dao động trên chip:

Chân 18 và 19.

- Các chân nguồn:

Chân 20 GND. Chân 40 VCC.

- RST (Reset):

Chân 9. Reset tích cực mức cao trong ít nhất 2 chu kỳ máy.

Các thanh ghi chức năng đặc biệt:

- Từ trạng thái chương trình:

Từ trạng thái chương trình (PSW: Program Status Word) ở địa chỉ D0H chứa các bit trạng thái như bảng tóm tắt sau:

Bảng 2.2: Chức năng các bit thanh ghi trạng thái chương trình

Bit	Ký hiệu	Địa chỉ	Mô tả bit
PSW.7	CY	D7H	Cờ nhớ
PSW.6	AC	D6H	Cờ nhớ phụ
PSW.5	F0	D5H	Cờ 0
PSW.4	RS1	D4H	Chọn dãy thanh ghi (bit 1)
PSW.3	RS0	D3H	Chọn dãy thanh ghi (bit 0) 00 = bank 0 : địa chỉ từ 00H đến 07H 01 = bank 1 : địa chỉ từ 08H đến 0FH 10 = bank 2 : địa chỉ từ 10H đến 17H 11 = bank 3 : địa chỉ từ 18H đến 1FH
PSW.2	OV	D2H	Cờ tràn
PSW.1	-	D1H	Dự trữ
PSW.0	P	D0H	Cờ kiểm tra chẵn lẻ

- Thanh ghi B:

Thanh ghi B ở địa chỉ F0H được dùng với thanh ghi tích lũy A cho phép toán nhân và chia. Lệnh MUL AB sẽ nhân các giá trị không dấu 8 bit trong A và B rồi trả về kết quả 16 bit trong A (byte thấp) và B (byte cao). Còn lệnh DIV AB sẽ chia A cho B rồi trả kết quả về kết quả nguyên trong A và phần dư trong B. Thanh ghi B cũng có thể được xem như thanh ghi đệm đa dụng. Nó được địa chỉ hóa từng bit bằng các địa chỉ bit FOH đến F7H.

- Con trỏ ngăn xếp:

Con trỏ ngăn xếp (SP) là một thanh ghi 8 bit ở địa chỉ 81H. Nó chứa địa chỉ của byte dữ liệu hiện hành trên đỉnh của ngăn xếp và lấy dữ liệu ra khỏi ngăn xếp. Lệnh cất dữ liệu vào ngăn xếp sẽ làm tăng SP trước khi ghi dữ liệu, và lệnh lấy dữ liệu ra khỏi ngăn xếp sẽ đọc dữ liệu và làm giảm SP.

- Con trỏ dữ liệu:

Con trỏ dữ liệu DPTR được dùng để truy xuất bộ nhớ ngoài là một thanh ghi 16 bit ở địa chỉ 82H (DPL: byte thấp) và 83H (DPH: byte cao).

- Các thanh ghi port xuất nhập:

Các Port của 89C52 bao gồm Port 0 ở địa chỉ 80H, Port 1 ở địa chỉ 90H, Port 2 ở địa chỉ A0H và Port 3 ở địa chỉ B0H. Tất cả các Port đều được địa chỉ hóa từng bit. Điều đó cung cấp một khả năng giao tiếp thuận lợi.

- Các thanh ghi định thời:

VĐK 89C52 chứa 3 bộ định thời đếm 16 bit được dùng trong việc định thời hoặc đếm sự kiện. Timer 0 ở địa chỉ 8AH (TL0: byte thấp) và 8DH (TH1: byte cao). Việc vận hành timer được set bởi thanh ghi Timer Mode (TMOD) ở địa chỉ 89H và thanh ghi điều khiển timer (TCON), còn ở địa chỉ 88H chỉ có TCON được địa chỉ hóa từng bit.

- Các thanh ghi port nối tiếp (SBUF)

Khi truyền dữ liệu thì ghi lên SBUF, khi nhận dữ liệu thì đọc SBUF. Các mode vận hành khác nhau được lập trình qua thanh ghi điều khiển port nối tiếp (SCON) (được địa chỉ hóa từng bit) ở địa chỉ 98H.

- Các thanh ghi ngắt:

89C52 có cấu trúc 6 nguồn ngắt, 2 mức ưu tiên. Các ngắt sau bị cấm sau khi reset hệ thống và sẽ được cho phép bằng việc ghi thanh ghi cho phép ngắt (IE) ở địa chỉ 8AH. Cả hai thanh ghi được địa chỉ hóa từng bit.

- Các thanh ghi điều khiển công suất:

Thanh ghi điều khiển công suất (PCON) ở địa chỉ 87H chứa nhiều bit điều khiển. Chúng được tóm tắt trong bảng sau:

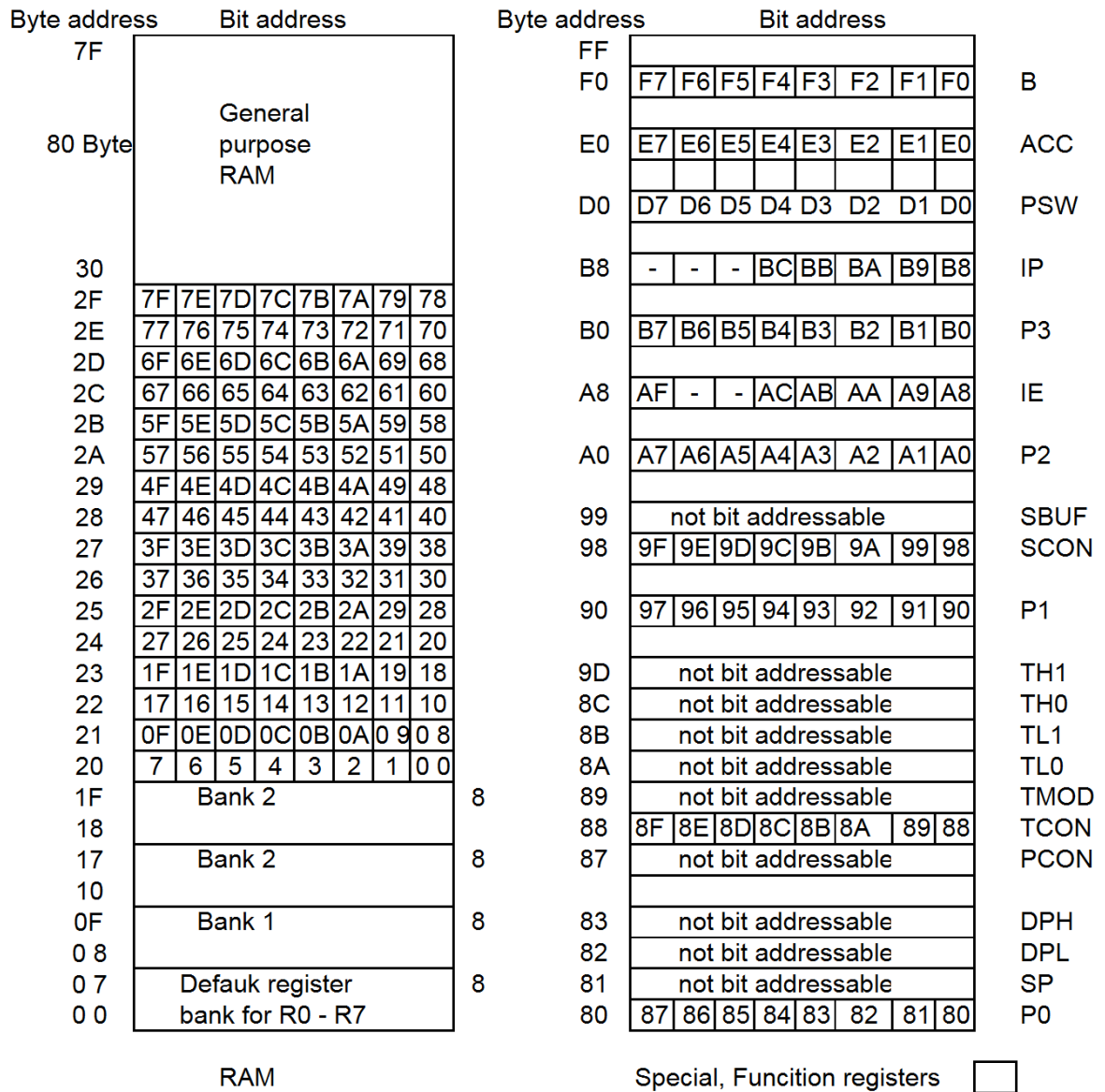
Bảng 2.3: Chức năng các bit thanh ghi điều khiển công suất

Bit	Ký hiệu	Mô tả
7	SMOD	Bit tăng gấp đôi tốc độ baud, bit này khi <i>set</i> làm cho tốc độ baud tăng 2 ở các chế độ 1, 2 và 3 của <i>port</i> nối tiếp.
6	-	Không định nghĩa
5	-	Không định nghĩa
4	-	Không định nghĩa
3	GF1	Bit cờ đa mục đích 1
2	GF0	Bit cờ đa mục đích 2
1	PD	Nguồn giảm; thiết lập để tích cực chế độ nguồn giảm, chỉ ra khỏi chế độ bằng reset.
0	ILD	Chế độ nghỉ; thiết lập để tích cực chế độ nghỉ, chỉ ra khỏi chế độ bằng 1 ngắt hoặc reset hệ thống.

Quan trọng nhất trong các thanh ghi trên là thanh ghi đếm chương trình, nó được đặt lại 0000H. Khi RST trở lại mức thấp, việc thi hành chương

trình luôn bắt đầu ở địa chỉ đầu tiên trong bộ nhớ chương trình: địa chỉ 0000H. Nội dung của RAM trên chip không bị thay đổi bởi lệnh reset.

Tổ chức bộ nhớ



Hình 2.5: Tổ chức bộ nhớ của 89C52

Mọi địa chỉ trong vùng RAM đa dụng đều có thể được truy xuất tự do, bằng cách đánh địa chỉ gián tiếp. Có 128 bit được địa chỉ hóa ở các byte từ 20H cho đến 2FH.

Hoạt động của bộ time

Bảng 2.4: Thanh ghi chức năng đặc biệt dùng time

SFR	Mục đích	Địa chỉ	Địa chỉ hóa từng bit
TCON	Điều khiển time	88H	Có
TMOD	Chế độ time	89H	Không
TL0	Byte thấp của time 0	8AH	Không
TL1	Byte thấp của time 1	8BH	Không
TH0	Byte cao của time 0	8CH	Không
TH1	Byte cao của time 1	8DH	Không

- Thanh ghi chế độ timer (TMOD)

Thanh ghi TMOD chứa 2 nhóm 4 bit dùng để đặt chế độ làm việc cho timer 0 và timer1.

Bảng 2.5: Các bit thanh ghi chế độ Time

Bit	Tên	Time	Mô tả
TMOD.7	GATE	1	Bit (mở) cổng, khi lên 1 time chỉ chạy khi INT 1 ở mức cao
TMOD.6	C/T	1	Bit chọn chế độ counter / time 1 : bộ đếm sự kiện 0 : bộ định khoảng thời gian

TMOD.5	M1	1	Bit 1 của chế độ (mode)
TMOD.4	M0	1	Bit 0 của chế độ 00 : chế độ 0 : time 13 bit 01 : chế độ 1 : time 16 bit 10 : chế độ 2 : tự động nạp lại 11 : chế độ 3 : tách time
TMOD.3	GATE	0	Bit (mở) cổng, khi lên 1 time chỉ chạy khi INT 1 ở mức cao
TMOD.2	C/T	0	
TMOD.1	M1	0	
TMOD.0	M0	0	

- Thanh ghi điều khiển timer (TCON)

Thanh ghi TCON chứa các bit trạng thái và các bit điều khiển cho timer 0 và timer 1.

Bảng 2.6: Tóm tắt thanh ghi TCON

Bit	Ký hiệu	Địa chỉ	Mô tả
TCON.7	TF1	8FH	Cờ tràn Time 1. Được đặt bởi phần cứng khi bộ Time 1 tràn. Được xóa bởi phần cứng khi bộ VXL hướng tới chương trình con phục vụ ngắt.
TCON.6	TR1	8EH	Bit điều khiển bộ Time 1 hoạt động. Được đặt / xóa bởi phần mềm để điều khiển bộ Time 1 ON / OFF.
TCON.5	TF0	8DH	Cờ tràn Time 0. Được đặt bởi phần cứng khi bộ Time 0 tràn. Được xóa bởi phần cứng khi bộ VXL hướng tới chương trình

			con phục vụ ngắt.
TCON.4	TR0	8CH	Bit điều khiển bộ Time 0 hoạt động. Được đặt / xóa bởi phần mềm để điều khiển bộ Time 0 ON / OFF.
TCON.3	IE1	8BH	Cờ ngắt ngoài 1. Được đặt bởi phần cứng khi sườn xung của ngắt ngoài 1 được phát hiện. Được xóa bởi phần cứng khi ngắt được xử lý.
TCON.2	IT1	8AH	Bit điều khiển ngắt 1 để tạo ra ngắt ngoài. Được đặt / xóa bởi phần mềm.
TCON.1	IE0	89H	Cờ ngắt ngoài 0. Được đặt bởi phần cứng khi sườn xung của ngắt ngoài 0 được phát hiện. Được xóa bởi phần cứng khi ngắt được xử lý.
TCON.0	IT0	88H	Bit điều khiển ngắt 0 để tạo ra ngắt ngoài. Được đặt / xóa bởi phần mềm.

Hoạt động port nối tiếp.

- Thanh ghi điều khiển port nối tiếp.

Chế độ hoạt động của port nối tiếp được đặt bằng cách ghi vào thanh ghi chế độ port nối tiếp (SCON) ở địa chỉ 98H. Sau đây các bảng tóm tắt thanh ghi SCON và các chế độ của port nối tiếp:

Bảng 2.7: Tóm tắt thanh ghi chế độ port nối tiếp SCON.

Bit	Ký hiệu	Địa chỉ	Mô tả
SCON.7	SM0	9FH	Bit 0 của chế độ chọn port nối tiếp
SCON.6	SM1	9EH	Bit 1 của chế độ chọn port nối tiếp
SCON.5	SM2	9DH	Bit 2 của chế độ 2 nối tiếp. Cho phép

			truyền thông đa xử lý trong các chế độ 2 và 3 ; R1 sẽ không bị tác động nếu bit thứ 9 thu được là 0.
SCON.4	REN	9CH	Cho phép bộ thu phải đặt lên 1 để thu các ký tự
SCON.3	TB8	9BH	Bit 8 phát, bit thứ 9 được phát các chế độ 2 và 3 : đặt được và xoá bằng phần mềm.
SCON.2	RB8	9AH	Bit 8 thu, bit thứ 9 thu được
SCON.1	TI	99H	Cờ ngắt phát. Đặt lên 1 khi kết thúc phát ký tự, được xoá bằng phần mềm.
SCON.0	RI	98H	Cờ ngắt thu. Đặt lên 1 khi kết thúc phát ký tự, được xoá bằng phần mềm.

Bảng 2.8: Các chế độ port nối tiếp.

SM0	SM1	Chế độ	Mô tả	Tốc độ baud
0	0	0	Thanh ghi dịch	$F_{OSC} / 12$
0	1	1	8 bit UART	Có thể thay đổi
1	0	2	9 bit UART	$F_{OSC} / 64$ hoặc $F_{OSC} / 32$
1	1	3	9 bit UART	Có thể thay đổi

Hoạt động ngắt.

- Cho phép và không cho phép ngắt.

Mỗi nguồn được cho phép hoặc không cho phép từng ngắt một qua thanh ghi chức năng đặc biệt cố định địa chỉ bit IE (Interrupt Enable: cho phép ngắt) ở địa chỉ A8H. Cũng như các bit cho phép mỗi nguồn ngắt, có một bit cho phép hoặc cấm an toàn bộ được xoá để cấm tất cả các ngắt hoặc được đặt lên 1 để cho phép tất cả các ngắt.

Bảng 2.9: Tóm tắt thanh ghi chế độ cho phép ngắt và không cho phép ngắt

Bit	Ký hiệu	Địa chỉ bit	Mô tả (1 = cho phép ;0 = cấm)
IE.7	EA	AFH	Cho phép hoặc cấm toàn bộ
IE.6	EA	AEH	Không được định nghĩa
IE.5	ET5	ADH	Cho phép ngắt từ Time 2
IE.4	E5	ACH	Cho phép ngắt Port nối tiếp
IE.3	ET1	ABH	Cho phép ngắt từ Time 1
IE.2	EX1	AAH	Cho phép ngắt ngoài 1
IE.1	ET0	A9H	Cho phép ngắt từ Time 0
IE.0	EX0	A8H	Cho phép ngắt ngoài 0

Ưu tiên ngắt.

Mỗi nguồn ngắt được lập trình riêng vào một trong hai mức ưu tiên qua thanh ghi chức năng đặc biệt được chỉ bit IP (Interrupt priority: ưu tiên ngắt) ở địa chỉ B8H.

Bảng 2.10: Tóm tắt thanh ghi IP.

Bit	Ký hiệu	Địa chỉ bit	Mô tả (1 = cho phép ;0 = cấm)
IP.7			Không được định nghĩa
IP.6			Không được định nghĩa
IP.5	PT2	BDH	Ưu tiên cho ngắt từ Time 2
IP.4	PS	BCH	Ưu tiên cho ngắt port nối tiếp
IP.3	PT1	BBH	Ưu tiên cho ngắt từ Time 1
IP.2	PX1	BAH	Ưu tiên cho ngắt ngoài 1
IP.1	PT0	B9H	Ưu tiên cho ngắt từ Time 0
IP.0	PX0	B8H	Ưu tiên cho ngắt ngoài 0

Các ngắt ưu tiên được xóa sau khi reset hệ thống để đặt tất cả các ngắt ở mức ưu tiên thấp hơn.

Xử lý ngắt.

Khi có một ngắt xảy ra và được CPU chấp nhận, chương trình chính bị ngắt quãng. Những hoạt động sau xảy ra:

- Thi hành hoàn chỉnh lệnh đang hiện hành
- Cất PC vào ngăn xếp
- Trạng thái ngắt hiện hành được cất bên trong
- Các ngắt được chặn tại mức của ngắt
- Nạp PC địa chỉ Vector của ISR
- ISR thực thi

ISR thực thi và đáp ứng ngắt. ISR hoàn tất bằng lệnh RETI. Điều này lấy lại giá trị cũ của PC từ ngăn xếp và lấy lại trạng thái ngắt cũ. Chương trình lại tiếp tục thi hành tại nơi mà nó dừng.

Các Vector ngắt

Khi chấp nhận ngắt, giá trị được nạp vào PC được gọi là vector ngắt. Nó là địa chỉ bắt đầu của ISR cho nguồn tạo ngắt. Các Vector ngắt được cho bởi bảng sau:

Bảng 2.11: Địa chỉ các vecter ngắt

Ngắt	Cờ	Địa chỉ vecter
Reset hệ thống	RST	0000H
Bên ngoài 0	IE0	0003H
Time 0	TF0	000BH
Bên ngoài 1	IE1	0013H
Time 1	TF1	001BH
Port nối tiếp	TI hoặc RI	0023H
Time 2	TF2 hoặc IE2	002BH

2.3. KHỐI HIỂN THỊ

2.3.1. Giới thiệu sơ lược về khối hiển thị

Để có thể biết được thời gian tín hiệu đèn giao thông có bao nhiêu giây thì cần phải có khối hiển thị thời gian báo cho người tham gia giao thông biết được thời gian của các đèn.

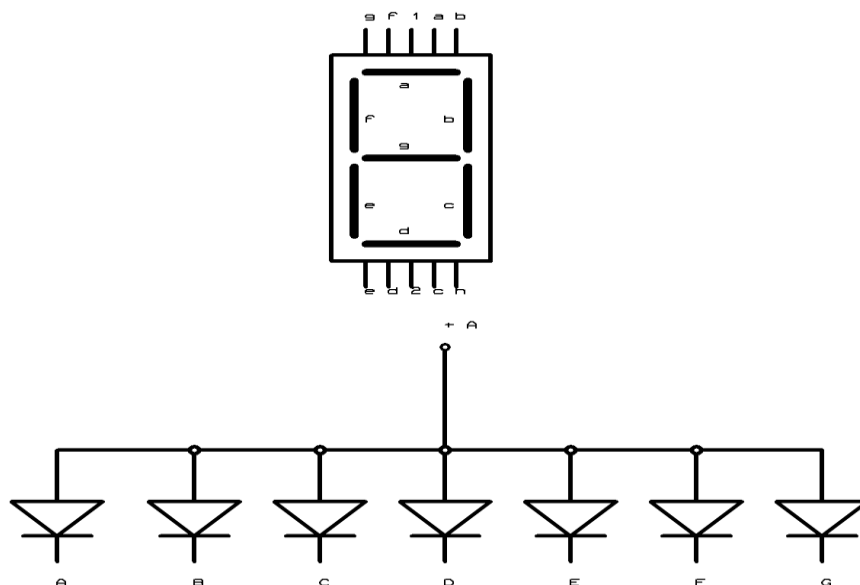
Khối hiển thị gồm có:

- 2 led 7 thanh ở mỗi cột để hiển thị thời gian của đèn TH tại ngã tư.
- 3 led đơn: 1 xanh, 1 đỏ, 1 vàng ở mỗi cột để hiển thị cho phép xe được đi hay xe phải dừng lại ở mỗi làn đường.
 - + Đèn xanh sáng cho phép xe được phép đi.
 - + Đèn đỏ sáng là các phương tiện phải dừng lại trước vạch kẻ sơn
- 4 IC ghi dịch 74HC595.

2.3.2. Led 7 thanh

Đối với các ứng dụng dùng hiển thị số liệu ra led 7 thanh, ta dùng mã hiển thị led 7 thanh ứng với mỗi loại led 7 đoạn (anode chung hay cathode chung) và tùy theo sơ đồ kết nối sẽ có một bảng mã riêng.

Với đề tài này chúng em sử dụng led 7 thanh loại Anode chung và có sơ đồ nguyên lý như sau:



Hình 2.6: Sơ đồ nguyên lý Led 7 thanh loại Anode chung.

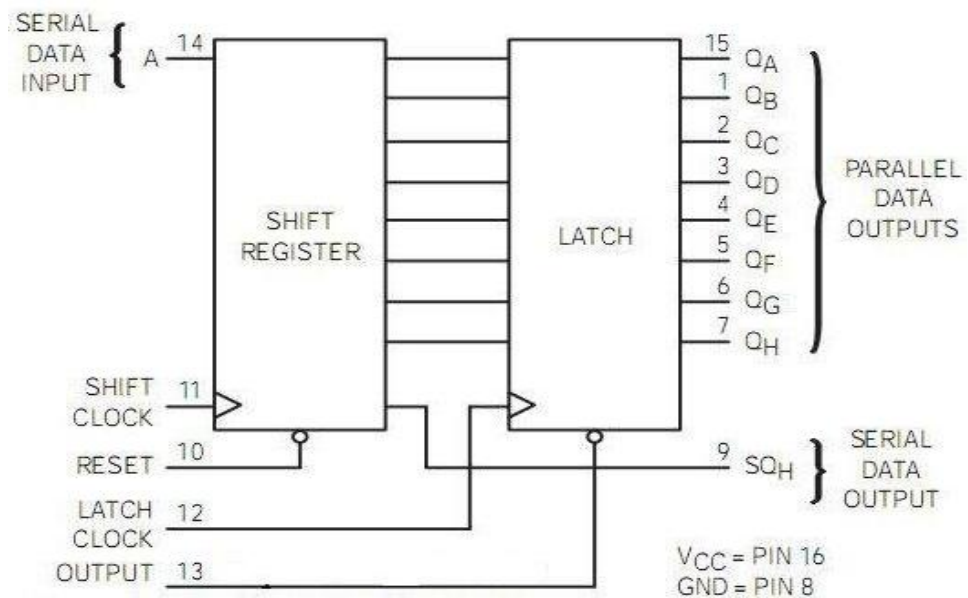
Mã led 7 đoạn loại Anode chung hiển thị số thập phân từ 0 đến 9 được thể hiện trong bảng sau:

Bảng 2.12: Bảng mã hiển thị Led 7 thanh

Số	P2.0 a	P2.1 b	P2.2 c	P2.3 d	P2.4 e	P2.5 f	P2.6 g	P2.7 h	P2
0	0	0	0	0	0	0	1	1	0C0H
1	1	0	0	1	1	1	1	1	0F9H
2	0	0	1	0	0	1	0	1	0A4H
3	0	0	0	0	1	1	0	1	0B0H
4	1	0	0	1	1	0	0	1	99H
5	0	1	0	0	1	0	0	1	92H
6	0	1	0	0	0	0	0	1	82H
7	0	0	0	1	1	1	1	1	0F8H
8	0	0	0	0	0	0	0	1	80H
9	0	0	0	0	1	0	0	1	90H

2.3.3. IC 74HC595

Sơ đồ hoạt động của chip :



Bảng thông số chip:

Symbol	Parameter	Value	Unit
V_{CC}	DC Supply Voltage (Referenced to GND)	- 0.5 to + 7.0	V
V_{in}	DC Input Voltage (Referenced to GND)	- 0.5 to $V_{CC} + 0.5$	V
V_{out}	DC Output Voltage (Referenced to GND)	- 0.5 to $V_{CC} + 0.5$	V
I_{in}	DC Input Current, per Pin	± 20	mA
I_{out}	DC Output Current, per Pin	± 35	mA
I_{CC}	DC Supply Current, V_{CC} and GND Pins	± 75	mA
PD	Power Dissipation in Still Air, Plastic DIP† SOIC Package† TSSOP Package†	750 500 450	mW
T_{stg}	Storage Temperature	- 65 to + 150	°C
T_L	Lead Temperature, 1 mm from Case for 10 Seconds (Plastic DIP, SOIC or TSSOP Package)	260	°C

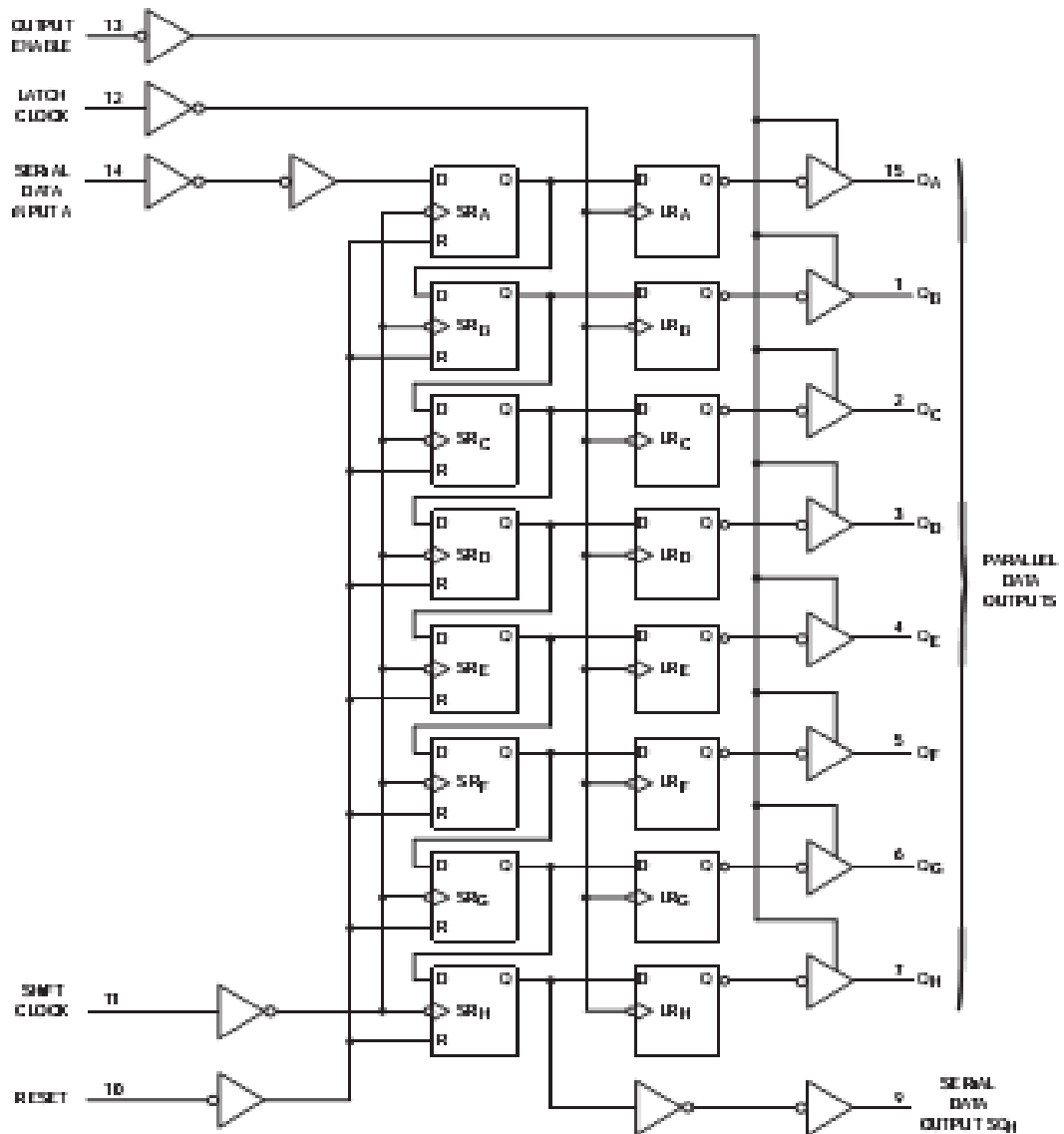
Đây là IC đầu ra hoạt động ở 2 mức đó là : mức **0** & **1** dòng ra tầm 35mA. Điện áp hoạt động $\leq 7V$. Công suất trung bình 500mW. Dựa vào bảng tính toán được các thông số khi thiết kế mạch:

Tần số đáp ứng:

Symbol	Parameter	Min	Max	Unit
V_{CC}	DC Supply Voltage (Referenced to GND)	2.0	6.0	V
V_{in}, V_{out}	DC Input Voltage, Output Voltage (Referenced to GND)	0	V_{CC}	V
T_A	Operating Temperature, All Package Types	- 55	+ 125	°C
t_r, t_f	Input Rise and Fall Time (Figure 1)	$V_{CC} = 2.0 V$ $V_{CC} = 4.5 V$ $V_{CC} = 6.0 V$	0 1000 500 400	ns

Tại 6V thì tần số vào đáp ứng khoảng 400ns . Dựa vào đó chúng ta sẽ đưa được ra tần số quét hợp lý.

IC 74HC595 có sơ đồ chân và hoạt động như sau:



Hình 2.7: Sơ đồ chân và nguyên tắc hoạt động của IC 74HC595.

IC 74HC595 có các chân và tác dụng hoạt động của chúng như sau:

- 8 chân xuất dữ liệu, chân **RESET (10)**, chân **cho phép chip (13)**.
- Một chân nhận dữ liệu vào nối tiếp (**14**): Ví dụ ta muốn dịch một bit **0** or **1** thì ta tác động vào đây.
- Một chân xung **Clock (11)**: Cứ mỗi xung tác động vào chân này thì bit logic tại chân nhận dữ liệu (**14**) sẽ được dịch vào một lần.

- Một chân **chốt dữ liệu (12)**: Mỗi lần có xung tác động vào chân này thì dữ liệu đc xuất ra một lần.

- Còn chân **(9)**: Thì cứ có 8 xung tác động vào chân clock nó lại xuất ra một xung.

- Kích một xung clock vào chân 11 rồi lại đưa chân 14 xuống mức logic 0 sau đó tác động 2 xung vào chân 11 rồi lại đưa chân 14 lên mức logic 1 tác động một xung vào chân 11 rồi lại đưa 14 xuống 0. Cuối cùng tác động một xung vào chân **(11)**, sau đó chốt bằng cách tác động một xung vào chân 12, nhìn trên Proteus ta sẽ thấy kết quả.

2.4. KHỐI NGUỒN

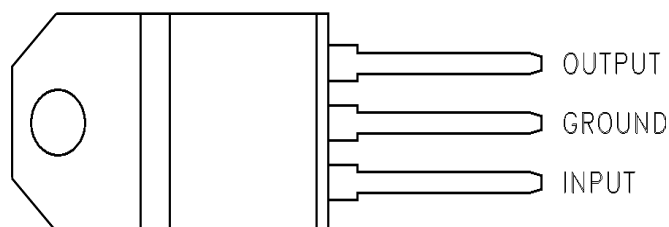
2.4.1. Giới thiệu sơ lược về khối nguồn

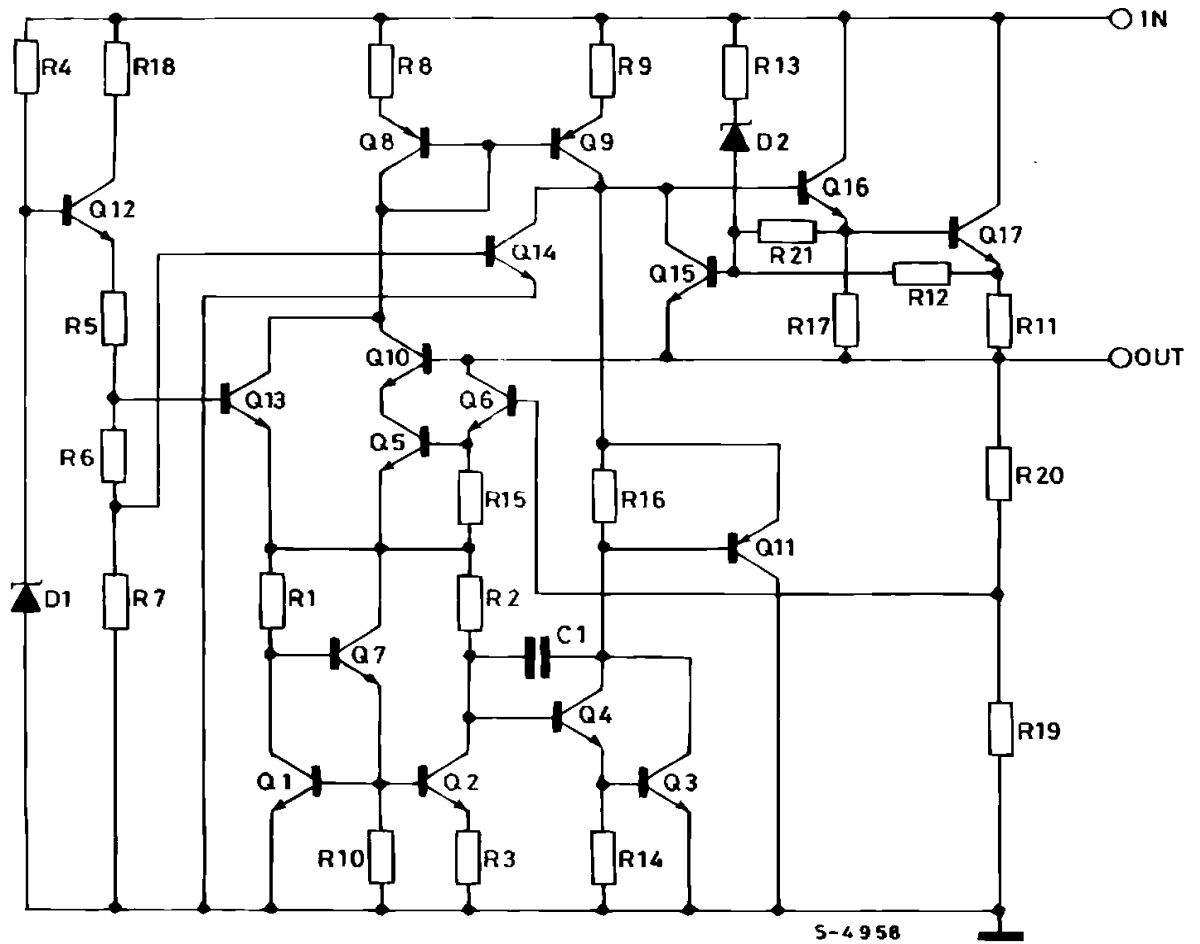
Để có điện áp cấp cho các khối cảm biến, khối xử lý trung tâm, khối hiển thị hay khối điều khiển động cơ ta cần có một khối nguồn có điện áp ra ổn định là +5V, vì khối nguồn có hoạt động ổn định thì các khối khác mới hoạt động ổn định được.

Với yêu cầu về điện áp như trên em sẽ xây dựng khối nguồn với IC chủ đạo là IC 7805 có điện áp lồi ra ổn định ở mức +5V.

2.4.2. IC 7805

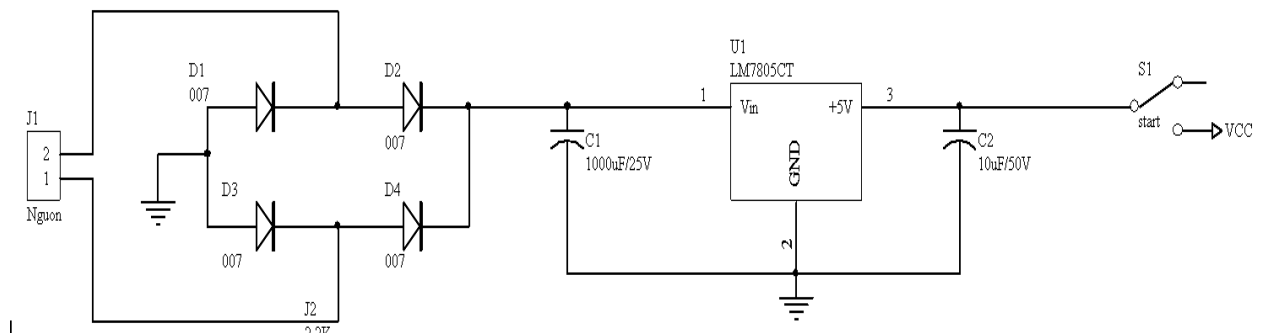
IC 7805 là loại IC ổn áp chuyên dụng để tạo ra điện áp ổn định +5V. IC 7805 có sơ đồ chân và cấu tạo như sau:





Hình 2.8: Sơ đồ chân và cấu tạo IC 7805

2.4.3. Sơ đồ nguyên lý khối nguồn



Hình 2.9: Sơ đồ nguyên lý khối nguồn

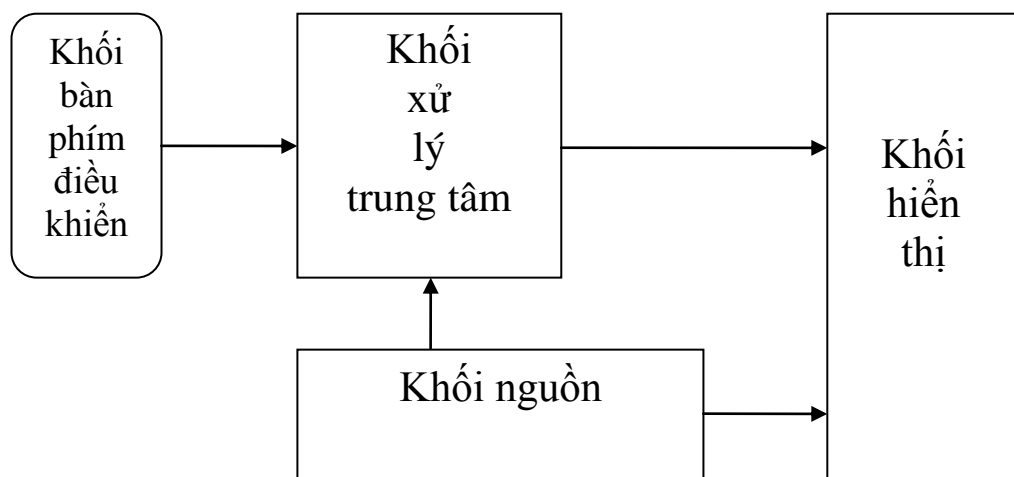
Theo như sơ đồ nguyên lý trên thì điện áp cấp cho khối nguồn có thể là AC hoặc DC vì đã qua chỉnh lưu cầu nhưng phải nhỏ hơn 30V vì V_{in} lớn nhất mà IC 7805 có thể hoạt động tốt với $U_{in} < +37V$.

CHƯƠNG 3.

XÂY DỰNG MÔ HÌNH VÀ THIẾT KẾ CHƯƠNG TRÌNH ĐIỀU KHIỂN

3.1. THIẾT KẾ VÀ THI CÔNG MẠCH ĐIỀU KHIỂN

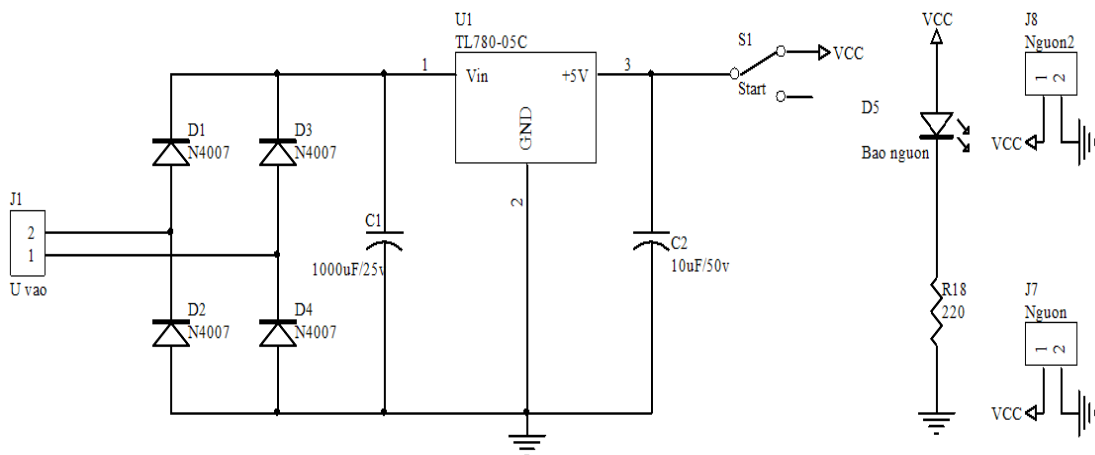
3.1.1. Xây dựng sơ đồ khối tổng quát



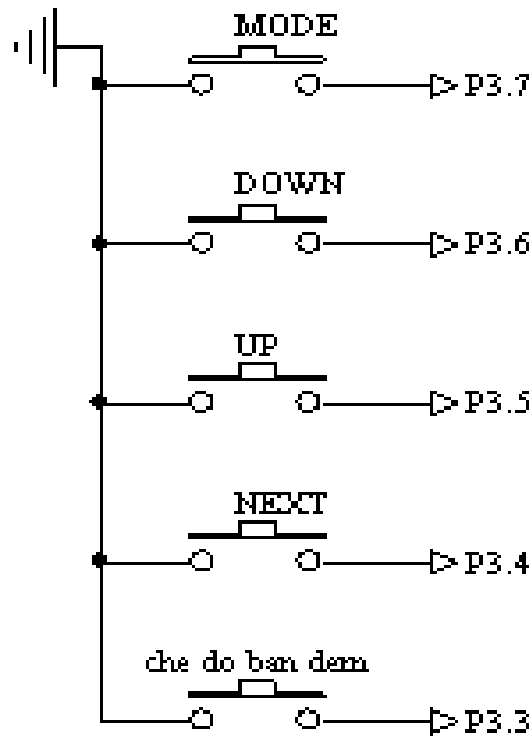
Hình 3.1: Sơ đồ khối tổng quát

3.1.2. Sơ đồ nguyên lý

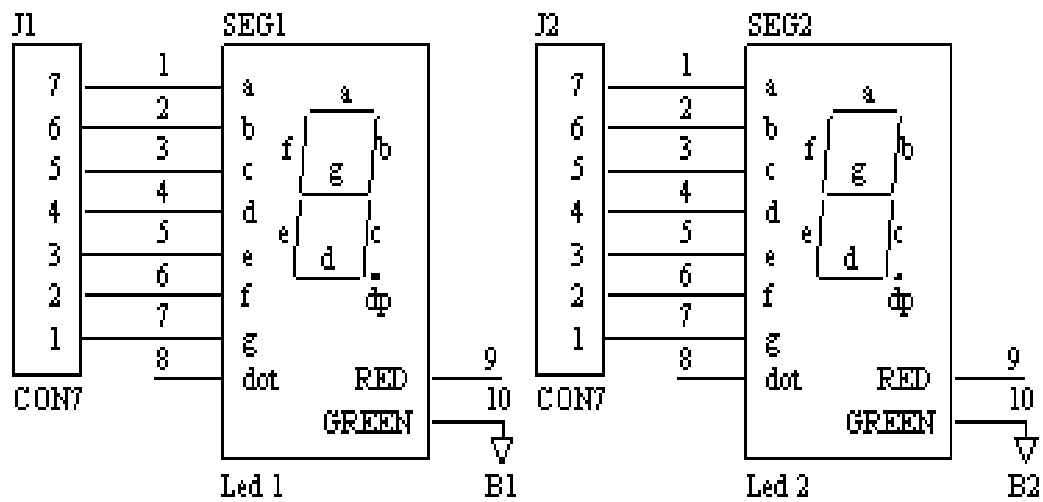
Trên cơ sở nghiên cứu các thiết bị sử dụng trong mô hình ta có thể xây dựng được sơ đồ nguyên lý mạch điều khiển như sau:



Hình 3.2: Sơ đồ khối nguồn



Hình 3.3: Sơ đồ khối phím điều khiển thời gian



Hình 3.4: Sơ đồ khối hiển thị

3.1.3. Nguyên lý hoạt động của đèn tín hiệu giao thông

Khi ta cấp nguồn cho mạch hoạt động, tín hiệu đèn **đỏ của cột 1** được xuất ra chân **P0.0**, và thời gian hiển thị của đèn đỏ là 30s, còn tín hiệu **đèn xanh cột 2** đc xuất ra chân **P0.5** hiển thị với thời gian là 26s. Sau khi đèn xanh cột 2 chạy hết 26s thì đèn xanh sẽ tắt, đèn vàng 2 sẽ bật lên hiển thị 4s, hết 4s đèn vàng 2 tắt. Tiếp đến là đèn đỏ cột 2 (P0.3) sẽ sáng trong vòng 30s, đèn xanh cột 1 sẽ sáng 26s, sau khi đèn xanh 1 đếm hết 26s thì đèn xanh 1 tắt và đèn vàng 1 được bật lên trong vòng 4s, sau 4s vàng 1 tắt, đèn sẽ lại nhảy về hiển thị ban đầu như trên cứ thế quay vòng. Như vậy ở đây ta sử dụng vòng lặp trong Code để dễ dàng lập trình cho mô hình.

Ngoài ra ta còn có 1 chế độ khác khá hay, chế độ này ta sử dụng các phím nhấn để điều khiển thay đổi thời gian. Ở mô hình nay ta có 4 phím nhấn:

- Mode (Thiết lập chế độ cài đặt thời gian) tức là khi ta nhấn Phím P3.7 toàn chương trình sẽ dừng lại chờ ta thiết lập thời gian bằng tay, đầu tiên ta thiết lập thời gian cho đèn đỏ cột 1 trước, tiếp đến là đèn xanh và cuối cùng sẽ là đèn vàng, ở đây ta chỉ cần thiết lập thời gian cho cột 1 còn lại thời gian ở cột 2 sẽ được cài đặt thời gian tương ứng với thời gian đã cài đặt ở cột 1. Sau khi cài đặt xong ta chỉ cần nhấn nút P3.7 thì chương trình sẽ hoạt động đúng với thời gian mà ta đã cài đặt.

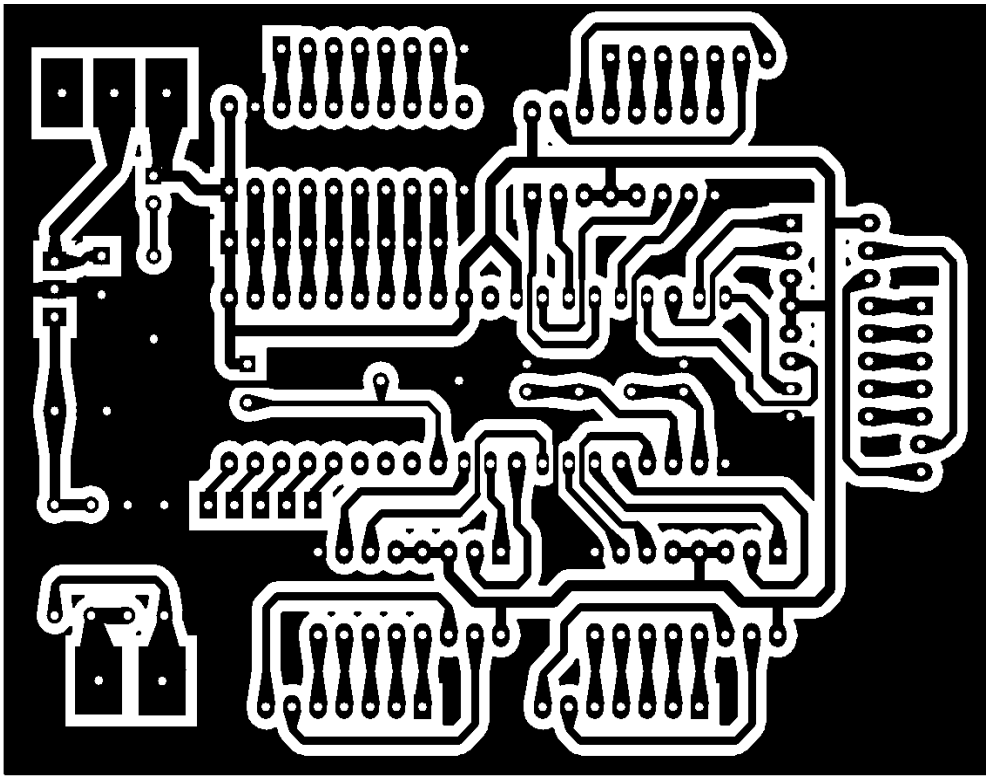
- Up (P3.6) nút này dùng để tăng thời gian của từng đèn.

- Down (P3.5) nút này dùng để giảm thời gian của từng đèn.

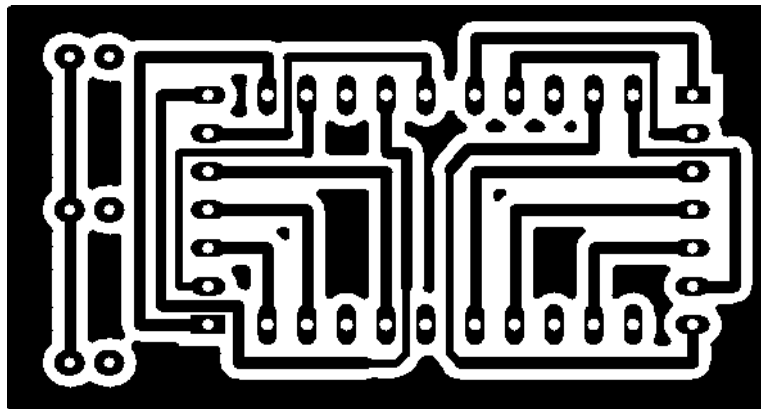
- Next (P3.4) nút này để chuyển từ đèn này sang đèn khác để thay đổi thời gian cho từng đèn.

3.1.4. Sơ đồ mạch in

Sau khi vẽ xong sơ đồ nguyên lý chúng ta sẽ chuyển sang sơ đồ mạch in bằng phần mềm vẽ mạch protel và tiến hành in mạch lắp ráp linh kiện.



Hình 3.6: Sơ đồ mạch in khối xử lý trung tâm



Hình 3.7: Sơ đồ mạch in khối hiển thị

3.1.5. Lắp ráp mô hình

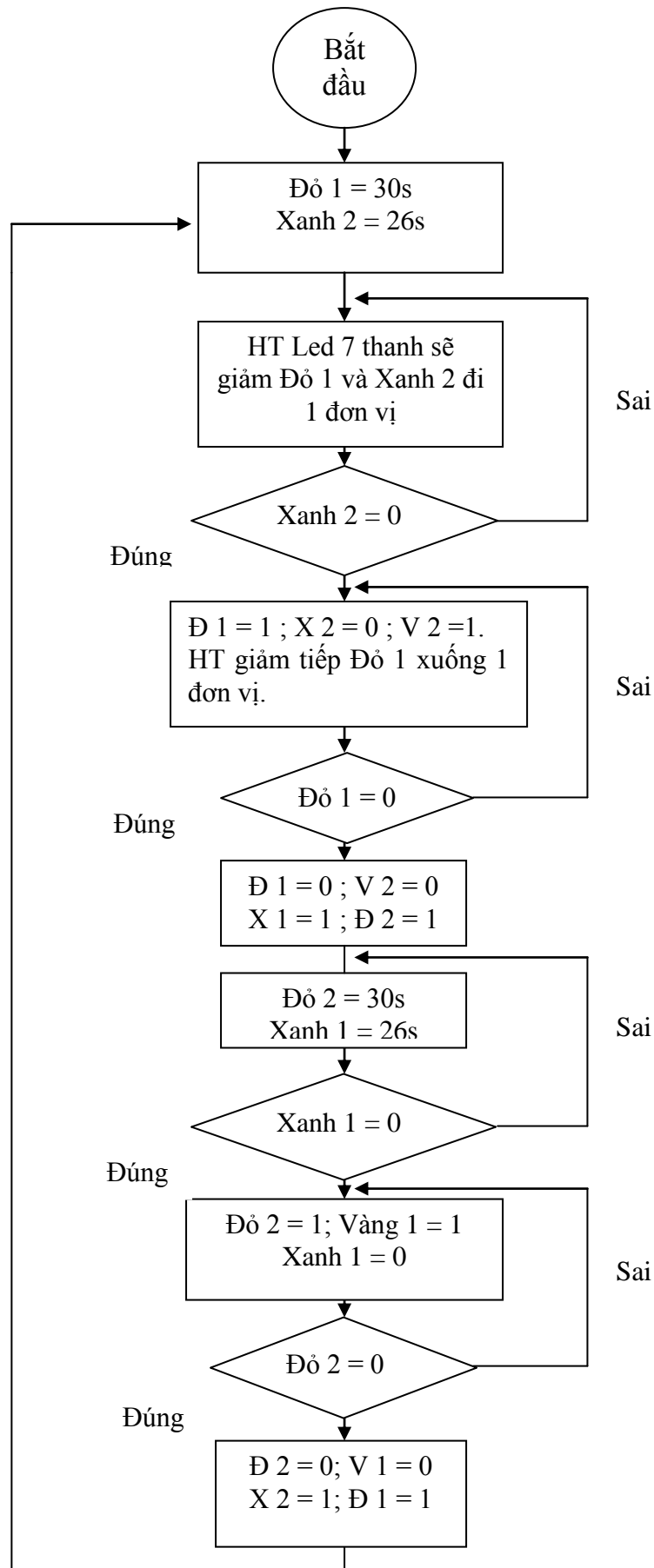
Sau khi đã làm xong mạch điều khiển chúng ta sẽ tiến hành làm công việc lắp ráp và chạy thử mô hình. Để thuận tiện cho việc thiết kế và lắp ráp ở mô hình này em sẽ dùng vật liệu là các tấm mê ca và cột trụ là các ống INOX để dễ dàng kết nối mô hình.

Mô hình sau khi hoàn thành:

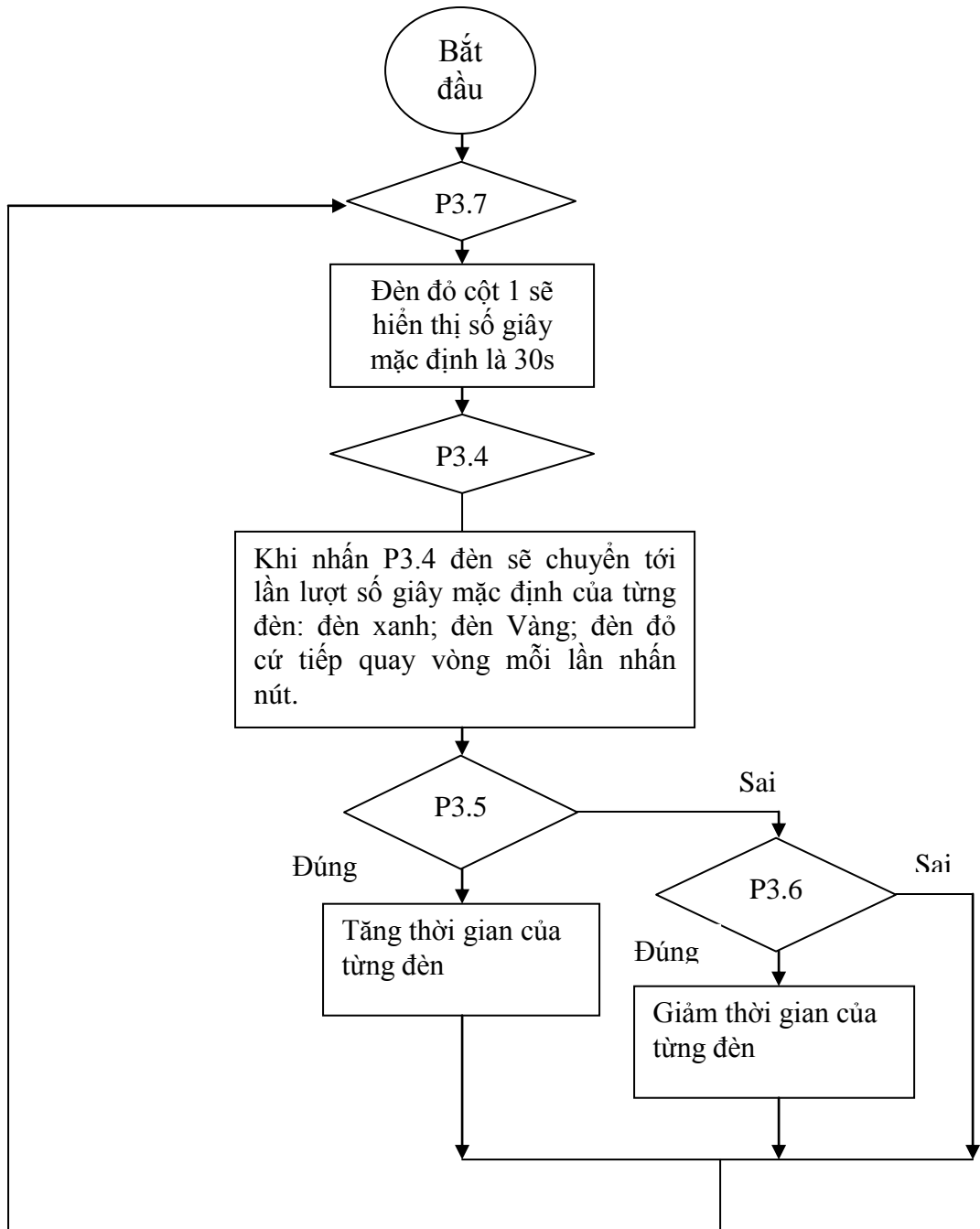


Hình 3.8: Mô hình đèn giao thông sau khi hoàn thành

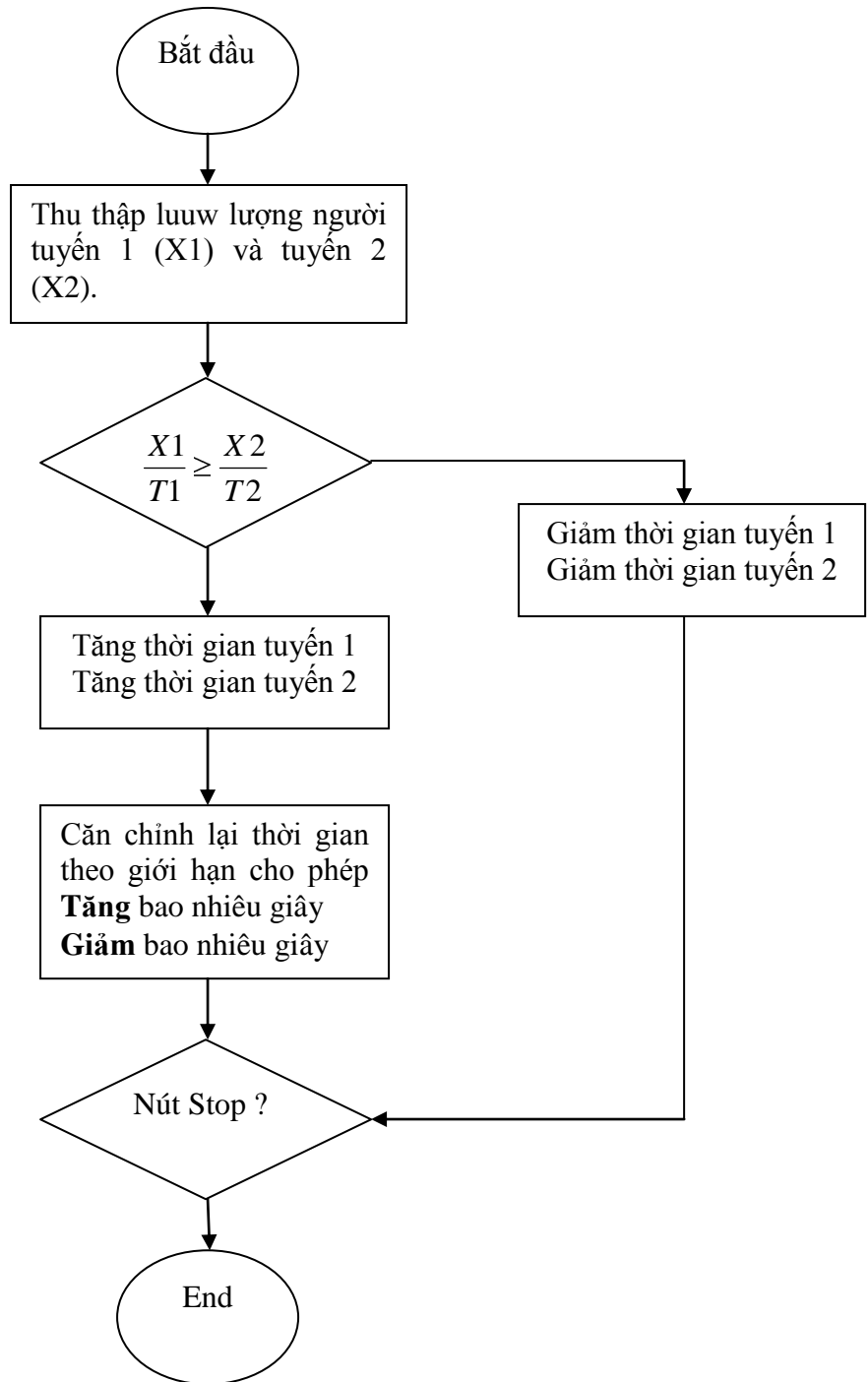
3.2. THUẬT TOÁN VÀ CHƯƠNG TRÌNH ĐIỀU KHIỂN



Hình 3.9: Lưu đồ chương trình hàm chính



Hình 3.10: Lưu đồ hàm điều chỉnh thời gian bằng tay.



Hình 3.11: Lưu đồ hàm tối ưu các tuyến theo lưu lượng

Bảng 3.1: Chú thích hoạt động các bit sử dụng trong mạch.

Bit	Chức năng hoạt động
P0.0	Điều khiển Led màu Đỏ 1
P0.1	Điều khiển Led màu Vàng 1
P0.2	Điều khiển Led màu Xanh 1
P0.3	Điều khiển Led màu Đỏ 2
P0.4	Điều khiển Led màu Vàng 2
P0.5	Điều khiển Led màu Xanh 2
P1.0 → P1.2	3 chân của VXL được nối với 3 chân tín hiệu đầu vào của IC 595 và đầu ra để điều khiển trạng thái LED 7 thanh.
P1.1 → P1.2	3 chân này cũng vậy lại được nối với một IC 595 khác và đầu ra điều khiển cho một LED 7 thanh khác.
P3.7	Mode (Thiết lập chế độ cài đặt thời gian).
P3.6	Giảm thời gian đèn tín hiệu giao thông
P3.5	Tăng thời gian đèn tín hiệu giao thông
P3.4	Chuyển đèn để thay đổi thời gian của từng đèn

3.2.2. Tính thời gian trễ

Để có thể điều khiển cho thời gian ở hai cột đèn giao thông chạy một cách chính xác thì cần phải tính toán thời gian sao cho. Vì thế ta sẽ sử dụng một chương trình con để chạy thời gian trễ, chương trình như sau:

```
tre:
                                mov  77h,#x
nhan1:                          mov  76h,#y
nhan2:                          mov  75h,#z
nhan3:                          djnz 75h,nhan3
                                djnz 76h,nhan2
                                djnz 77h,nhan1
                                ret
```

Trong đó: x,y,z là các số nguyên dương

Công thức tính thời gian như sau:

$$t = [(77h * 2 + 2) * 76h + 2] * 75h + 2 + 2 + 2 (\mu s)$$

3.2.3. Chương trình điều khiển

```
$include(reg51.inc)
org 0000h
luudl:
mov dptr,#dl
mov p3,#0ffh
mov p0,#0ffh
mov r6,#0;o nho thoi gian den xanh2(54h)
mov r7,#0;o nho thoi gian den do2(55h)
mov r0,#0
mov r1,#0
mov r4,#0
```

```

mov r5,#0
mov r3,#0
mov 64h,#0
mov 51h,#10      ;do1:(=xanh2+vang2)
mov 52h,#12      ;xanh1:( = do2 - vang )
mov 53h,#4       ;vang:(vang=vang1=vang2)
mov r5,#0
mov r3,#0
start:
quyetbanphim:
    jmp chuong_trinh_chinh
hientinhapthoigianchinh:
    mov p0,#0ffh
    lcall tadcotden2
hientinhapthoigian:
    ;=====
htdendo1:
    cjne r5,#0,htdenxanh1
    clr p0.7;bat den do1
    setb p0.6;tat den xanh1
    setb p0.5;tat den vang1
    mov 57h,51h
lapdendo1:
    call timgiatrir0
    call hienthinaptg
    mov 79h,51h
    call tanggiamthoigian
    mov 51h,79h

```

;=====

htdenxanh1:

```
cjne r5,#1,htdenvang1
clr p0.6;bat den do1
setb p0.7;tat den xanh1
setb p0.5;tat den vang1
mov 57h,52h
```

lapdenxanh1:

```
call timgiatrir0
call hienthinaptg
mov 79h,52h
call tanggiamthoigian
mov 52h,79h
```

;=====

htdenvang1:

```
cjne r5,#2,ktpheimnex
clr p0.5;bat den do1
setb p0.7;tat den xanh1
setb p0.6;tat den vang1
mov 57h,53h
```

lapdenvang1:

```
call timgiatrir0
call hienthinaptg
mov 79h,53h
call tanggiamthoigian
mov 53h,79h
```

;=====

ktpheimnex:

```

    mov 66h,#40
    ;=====
ktphimnexdcn:
    jb p3.4,ktphimmenudcan_1
    djnz 66h,ktphimnexdcn
ktphimnexdcnha:
    mov 66h,#40
ktphimnexdcnha_1:
    jnb p3.4,ktphimnexdcnha
    djnz 66h,ktphimnexdcnha_1
    inc r5
    cjne r5,#3,kiemtraphimmenudcanlan2
    mov r5,#0
kiemtraphimmenudcanlan2:
    mov 66h,#40
    ;=====
ktphimmenudcan_1:
    jb p3.7,hienthinhapthoigian
    djnz 66h,ktphimmenudcan_1
ktphimmenudcnha_11:
    mov 66h,#40
ktphimmenudcnha_1:
    jnb p3.7,ktphimmenudcnha_11
    djnz 66h,ktphimmenudcnha_1
;=====
CHAYLAI:
    mov dptr,#dl
    mov p3,#0ffh

```

```

mov p0,#0ffh
mov r6,#0;o nho thoi gian den xanh2(54h)
mov r7,#0;o nho thoi gian den do2(55h)
mov r0,#0
mov r1,#0
mov r4,#0
mov r5,#0
mov r3,#0
mov 64h,#0

;=====
chuong_trinh_chinh:
    call timgiatrir6_r7
chuongtrinh1:
    clr p0.7;bat den do1
    clr p0.3;bat den xanh2
    mov 57h,51h
    mov 58h,54h
dendo1:
    call timgiatrir0_r1
dendo1_1:
    call hienthi
    call doi
    dec 57h;dendo
    jb p0.3,denvang2;neu den xanh dc tat thi nhay xuong den vang2
    cjne r4,#0,nhaycoc1
    djnz 58h,dendo1;denxanh2
nhaycoc1:
    inc r4

```



```

    cjne r4,#1,chuyentiep1
    jmp dendo1
chuyentiep1:
    mov r4,#0
    inc 78h
    mov r5,78h
    cjne r5,#1,denvang2
    jmp dendo1
denvang2:
    jnb p0.2,chaydenvang2
    setb p0.3;tat den xanh2
    clr p0.2;bat den vang2
    mov 58h,53h
chaydenvang2:
    cjne r4,#0,nhaycoc2
    djnz 58h,dendo1
nhaycoc2:
    inc r4
    cjne r4,#1,chuongtrinh2
    jmp dendo1
;=====
chuongtrinh2:
    mov r4,#0
    setb p0.7;tat den do1
    setb p0.2;tat den vang1
    clr p0.4;bat den do2
    clr p0.6;bat den xanh1
    mov 57h,52h

```

```

    mov 58h,55h
dendo2:
    call timgiatrir0_r1
dendo2_1:
    call hienthi
    call doi
    dec 58h;dendo
    jnb p0.6,denvang1;neu den xanh1 dc tat thi nhay xuong den vang1
    cjne r4,#0,nhaycoc1_1
    djnz 57h,dendo2;denxanh2
nhaycoc1_1:
    inc r4
    cjne r4,#1,chuyentiep2
    jmp dendo2
chuyentiep2:
    mov r4,#0
    inc 78h
    mov r5,78h
    cjne r5,#1,denvang1
    jmp dendo2
denvang1:
    jnb p0.5,chaydenvang1
    setb p0.6;tat den xanh1
    clr p0.5;bat den vang1
    mov 57h,53h;nap thoi gian cho den vang1
chaydenvang1:
    cjne r4,#0,nhaycoc2_1
    djnz 57h,dendo2

```

```

nhaycoc2_1:
    inc r4
    cjne r4,#1,chuongtrinh3
    jmp dendo2
chuongtrinh3:
    jmp CHAYLAI
;=====
tanggiamthoigian:
ktphimgiam:
    mov 66h,#50
ktphimgiam_1:
    jb p3.6,kiemtraphimtang
    djnz 66h,ktphimgiam_1
kiemtraphimgiam2:
    mov 66h,#50
kiemtraphimgiam2_1:
    jnb p3.6,kiemtraphimgiam2
    djnz 66h,kiemtraphimgiam2_1
    djnz 79h,thoat_tanggiam
    mov 79h,#99
    jmp thoat_tanggiam
kiemtraphimtang:
    mov 66h,#50
kiemtraphimtangan:
    jb p3.5,thoat_tanggiam
    djnz 66h,kiemtraphimtangan
    mov 66h,#50
kiemtraphimtangnha:

```

```

jnb p3.5,kiemtraphimtangnha
djnz 66h,kiemtraphimtangnha
inc 79h
mov r3,79h
    cjne r3,#100,thoat_tanggiam
    mov 79h,#0
thoat_tanggiam:
    ret

;=====
hienthinaptg:
    mov 75h,#2
hienthinaptg_1:
    mov a,r0
    movc a,@a+dptr
    mov 50h,#8
lap1_a:
    rlc a
    mov p2.0,c;dua dl vao
    setb p2.2;xung ck
    nop
    clr p2.2;xung ck
    djnz 50h,lap1_a
    dec r0
    djnz 75h,hienthinaptg_1
    setb p2.1;chot
    nop
    clr p2.1;chot
    ret

```

;=====

tingiattrir0:

laythoigiancot1_1:

mov r0,#0

mov 60h,57h

lay1_a:

inc r0

inc r0

djnz 60h,lay1_a

inc r0

ret

;=====

tatcotden2:

clr p1.2;dua dl vao

mov 50h,#16

lap2_2:

setb p1.4;xung ck

nop

clr p1.4;xung ck

djnz 50h,lap2_2

setb p1.3;chot

nop

clr p1.3;chot

ret

;=====

hienthi:

hienthicot1:

mov 75h,#2

```

hienthicot1_1:
    mov a,r0
    movc a,@a+dptr
    mov 50h,#8
lap1: rlc a
    mov p2.0,c;dua dl vao
    setb p2.2;xung ck
    nop
    clr p2.2;xung ck
    djnz 50h,lap1
    dec r0
    djnz 75h,hienthicot1_1
    setb p2.1;chot
    nop
    clr p2.1;chot
hienthicot2:
    mov 75h,#2
hienthicot2_1:
    mov a,r1
    movc a,@a+dptr
    mov 50h,#8
lap2: rlc a
    mov p1.2,c;dua dl vao
    setb p1.4;xung ck
    nop
    clr p1.4;xung ck
    djnz 50h,lap2
    dec r1

```

```

    djnz 75h,hienthicot2_1
    setb p1.3;chot
    nop
    clr p1.3;chot
    ret

;=====

timgiatrir0_r1:
laythoigiancot1:
    mov r0,#0
    mov 60h,57h
lay1: inc r0
    inc r0
    djnz 60h,lay1
    inc r0
laythoigiancot2:
    mov r1,#0
    mov 60h,58h
lay2: inc r1
    inc r1
    djnz 60h,lay2
    inc r1
    ret

;=====

timgiatrir6_r7::(tim thoi gian den xanh2 va den do2)
timthoigiandenxanh2::(=deno1-denvang)
    mov 56h,51h
giam1:
    dec 56h

```

```

    inc r6
    mov a,56h
    cjne a,53h,giam1
    mov 54h,r6
timthoigiandendo2:
    mov 56h,52h
giam2:
    inc r7
    djnz 56h,giam2
    mov 56h,53h
giam3: inc r7
    djnz 56h,giam3
    mov 55h,r7
    ret

;=====
doi:
    mov 77h,#2
nhan3: mov 76h,#200
nhan1: mov 75h,#250
nhan:
phimmenu:
    mov 66h,#40
ktphimmenu:
    jb p3.7,kiemtrahienthibandem
    djnz 66h,ktphimmenu
ktphimmenu1:
    mov 66h,#40
ktphimmenu2:

```



```

jnb p3.7,ktphimmenucan1
djnz 66h,ktphimmenucnha
jmp hienthinhapthoigianchinh
kiemtrahienthibandem:
    jnb p3.0,chaybinhthuong
chaybinhthuong:
    djnz 75h,nhan
    djnz 76h,nhan1
    djnz 77h,nhan3
    ret
;=====
dl:
    DB 7EH,7EH; 00
    DB 7EH,30H; 01
    DB 7EH,6DH; 02
    DB 7EH,79H; 03
    DB 7EH,33H; 04
    DB 7EH,5BH; 05
    DB 7EH,5FH; 06
    DB 7EH,70H; 07
    DB 7EH,7FH; 08
    DB 7EH,7BH; 09
END

```

KẾT LUẬN

Trong khoảng thời gian ba tháng làm đồ án em đã học hỏi được thêm nhiều kiến thức quan trọng trong lĩnh vực điện tự động và đã hoàn thành đề tài mà giáo viên hướng dẫn giao cho.

Những việc đã làm được:

- + Đã hoàn mô hình, chương trình điều khiển đèn tín hiệu giao thông tại ngã tư.
- + Mô hình chạy đúng theo yêu cầu mà giáo viên hướng dẫn đề ra đó là thực hiện việc điều khiển và thay đổi được thời gian theo lưu lượng người.

Những việc chưa làm được:

- + Mô hình đèn giao thông có tính thẩm mỹ chưa cao.
- + Hệ thống tự động hóa chưa được cao.

Trong hai năm học tập tại trường. Em đã được sự hướng dẫn tận tình của các thầy cô trong khoa về những kiến thức chuyên môn cũng như kiến thức trong cuộc sống. Từ những kiến thức nền tảng đó đã giúp em hoàn thành tập đồ án tốt nghiệp này.

Em xin chân thành cảm ơn thầy cô trong khoa “điện công nghiệp” đã giảng dạy cho chúng em những kiến thức về chuyên môn và định hướng đi theo sự hiểu biết, khả năng của chúng em để chúng em thực hiện tốt đề án và tạo điều kiện thuận lợi cho chúng em hoàn tất khóa học cũng như công việc sau này. Sau cùng em xin chân thành cảm ơn thầy *Ths. Nguyễn Trọng Thắng* đã tận tình giúp đỡ em hoàn thành tập đề án này.

Em xin chân thành cảm ơn!

Hải Phòng, ngày tháng năm 2012

Sinh viên

Nguyễn Mạnh Tùng

TÀI LIỆU THAM KHẢO

1. PGS-TS Nguyễn Tiến Ban(2010), *Bài giảng Phần tử tự động*, Trường Đại học Hàng Hải Việt Nam
2. Đặng Văn Đào – Lê Văn Doanh (2006), *Giáo trình Kỹ thuật điện*, Nhà Xuất Bản khoa học và kỹ thuật Hà Nội
3. Nguyễn Văn Hòa (2008), *Giáo trình Đo lường điện và cảm biến đo lường*, Nhà xuất bản giáo dục
4. Tống Văn On – Hoàng Đức Hải (2004), *Họ vi điều khiển 8051*, Nhà xuất bản Lao động – Xã hội