TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP.HCM
KHOA ĐIỆN – ĐIỆN TỬ VIỄN THÔNG

HUỲNH TUẤN ANH - DV02002

THIẾT KẾ NGÔI NHÀ THÔNG MINH

ĐỒ ÁN TỐT NGHIỆP

GIÁO VIÊN HƯỚNG DẪN :

NGUYỄN THỊ NGỌC ANH

TRƯỜNG ĐẠI HỌC GTVT TP.HCM
THƯ VIỆN
981
ĐV 18/2007

NIÊN KHÓA 2002 – 2007

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TPHCM
KHOA ĐIỆN - ĐIỆN TỬ VIỄN THÔNG

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc Lập - Tự Do - Hạnh Phúc
- - - o O o - - -

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

(Chú ý: SV phải đóng bản nhiệm vụ này vào trang thứ nhất của đồ án)

Họ và tên SV: Huỳnh Tuấn Anh ..... MSSV: D.V.02.002 .....

Ngành: Điện Tử - Viễn Thông ..... Lớp: 242 .....

1. **Tên đồ án tốt nghiệp:**

.....
Khái niệm ngôi nhà thông minh - Thiết kế các hệ điều khiển
.....
.....
.....

2. **Nhiệm vụ (yêu cầu về nội dung và số liệu ban đầu):**

.....
Tìm hiểu khái niệm ngôi nhà thông minh
Tìm hiểu vi điều khiển 89C51
Tìm hiểu các hệ thống cảm biến ánh sáng, nhiệt độ
Thiết kế thi công hệ thống ngôi nhà thông minh
.....
.....
.....
.....
.....

3. **Ngày giao nhiệm vụ đồ án:** 23/04/2007

4. **Ngày hoàn thành nhiệm vụ:** 23/07/2007

5. **Họ tên người hướng dẫn:** **Phần hướng dẫn**

1) Th.s. Nguyễn Thị Ngọc Anh ..... 1) .....
2) ..... 2) .....
3) ..... 3) .....

Nội dung và yêu cầu ĐATN đã được thông qua.
Ngày 04 tháng 08 năm 2007

TRƯỞNG KHOA
(Ký và ghi rõ họ tên)

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TP. HỒ CHÍ MINH

PGS. TS. Trần Hoài An

NGƯỜI HƯỚNG DẪN CHÍNH
(Ký và ghi rõ họ tên)

Nguyễn Thị Ngọc Anh

Lời cảm ơn

Có lẽ không một ai có thể quên được những mái trường mà mình đã đi qua trong một thời thơ ấu cho đến lúc trưởng thành. Cứ sau mỗi chặng đường đi qua, chúng em lại thấy mình lớn lên, vững vàng hơn trong kiến thức và năng lực. Chặng đường vừa đi qua là chặng đường với 5 năm dài đầy gay go và thử thách dưới mái Trường Đại học Giao Thông Vận Tải Tp. Hồ Chí Minh. Nơi đây đã không chỉ đơn giản cung cấp cho chúng em những kiến thức khoa học kỹ thuật mà quý hơn cả là đã nhóm trong chúng em ngọn lửa yêu khoa học và rèn luyện cho chúng em một nghị lực vững vàng để làm hành trang bước vào đời.

Xin cảm ơn những mái trường mà em đã đi qua suốt thời niên thiếu. Em sẽ ghi nhớ mãi công ơn của tất cả thầy cô đã dìu dắt em từ những buổi học vỡ lòng đến nay.

Kính lời cảm ơn khoa Điện- Điện tử Viễn Thông cùng các thầy cô của khoa, đã dành nhiều tình cảm tốt đẹp đối với chúng em và tạo mọi điều kiện thuận lợi để chúng em học tập tốt.

Gởi đến cô Nguyễn Thị Ngọc Anh lời cảm ơn chân thành, người đã trực tiếp hướng dẫn em trong suốt quá trình làm đồ án.

Bên cạnh đó gia đình còn là nguồn động lực to lớn về tinh thần lẫn vật chất giúp cho chúng em bước đi trên con đường mà chúng em đã chọn. Xin chân thành cám ơn Cha, Mẹ và chị Hai đã hết sức động viên và giúp đỡ em về mọi mặt trong suốt quá trình học tập.

Sau cùng là lời cảm ơn chân thành về sự giúp đỡ và những ý kiến đóng góp của các bạn cùng lớp trong suốt quá trình làm đề tài.

Tp HCM, tháng 7 năm 2007

Sinh viên thực hiện

Huỳnh Tuấn Anh

MỤC LỤC

Lời cảm ơn

Tóm tắt luận văn

Lời mở đầu

Trang

PHẦN 1. TỔNG QUAN VỀ NGÔI NHÀ THÔNG MINH 1

1. Khái niệm 1
2. Các chuẩn cho công nghệ nhà thông minh 1
   2.1.1. Chuẩn X10 1
      2.1.1.1. Giới thiệu 1
      2.1.1.2. Nguyên lý hoạt động 1
      2.1.1.3. Bộ điều khiển 3
      2.1.1.4. Thiết bị được điều khiển 3
      2.1.1.5. Mô hình thực tế 3
   2.1.2. Chuẩn INSTEON 4
      2.1.2.1. Giới thiệu 4
      2.1.2.2. Nguyên lí hoạt động 5
      2.1.2.3. Mô hình thực tế 6
3. Các hệ thống trong ngôi nhà thông minh 7
4. Giới hạn nhiệm vụ đồ án 8

PHẦN 2. KIẾN THỨC HỖ TRỢ 9

1. Cảm biến 9
   1.1. Cảm biến ánh sáng – Quang trở 9
      1.1.1. Mô tả hoạt động .....
      1.1.2. Các đặc tuyến của quang trở 10

1.2. Cảm biến nhiệt độ 11
1.2.1. Nhiệt điện trở kim loại 11
1.2.1.1. Mô tả hoạt động 11
1.2.1.2. Sơ đồ đo 13
1.2.2. Nhiệt điện trở bán dẫn 14
1.2.2.1. Mô tả hoạt động 14
1.2.2.2. Sơ đồ đo 14
1.2.3. IC chuyên dụng 15
2. IC chuyển đổi số - tương tự ADC 0809 16
2.1. Giới thiệu IC 16
2.2. Đặc điểm, sơ đồ chân và chức năng 16
2.3. Nguyên tắc hoạt động 17
3. Vi điều khiển 89C51 18
3.1. Giới thiệu IC 18
3.2. Sơ đồ chân và sơ đồ các khối bên trong 19
3.3. Chức năng các chân 20
3.4. Giới thiệu port nối tiếp và thanh ghi điều khiển SCON 21
3.5. Các chế độ hoạt động của port nối tiếp 23
3.5.1. Chế độ 0 –Thanh ghi dịch 8 bit 24
3.5.2. Chế độ 1 –UART 8 bit tốc độ thay đổi 24
3.5.3. Chế độ 2 –UART 9 bit tốc độ cố định 25
3.5.4. Chế độ 3 –UART 9 bit tốc độ thay đổi 26
4. Thiết bị hiển thị LCD 28
4.1. Giới thiệu thiết bị 28
4.2. Sơ đồ chân và chức năng 28
4.3. Các mã lệnh và trình tự điều khiển 28
5. IC ứng dụng trong điều khiển từ xa 29
5.1. Giới thiệu IC SC9148 29
5.1.1. Sơ đồ chân và chức năng 30

5.1.2. Sơ đồ mạch ứng dụng và nguyên lý hoạt động 30
5.2. Giới thiệu IC SC9149 32
5.2.1. Sơ đồ chân và chức năng 32
5.2.2. Sơ đồ mạch ứng dụng và nguyên lý hoạt động 33
6. Giao tiếp máy tính và 89C51 34
6.1. Giới thiệu chuẩn giao tiếp RS-232 34
6.2. Giới thiệu IC MAX 232 34
6.3. Đầu nối và sơ đồ kết nối 35

PHẦN 3. TÍNH TOÁN – THIẾT KẾ MẠCH 37
1. Mạch điều khiển cửa 37
1.1. Sơ đồ khối và chức năng các khối 37
1.2. Tính toán thiết kế 39
1.2.1. Khối bàn phím : 39
1.2.2. Khối hiển thị LCD 40
1.2.3. Khối trạng thái thiết bị 40
1.2.4. Khối nút điều khiển 40
1.2.5. Khối motor 41
2. Mạch điều khiển trung tâm 44
2.1. Sơ đồ khối và chức năng các khối 44
2.2. Tính toán thiết kế 45
2.2.1. Khối cảm biến nhiệt độ 45
2.2.2. Khối cảm biến ánh sáng 47
2.2.3. Khối chuyển đổi analog-digital 48
2.2.3.1. Mạch tạo điện áp chuẩn 48
2.2.3.2. Mạch tạo xung clock 49
2.2.4. Khối vi điều khiển 50
2.2.4.1. Mạch reset hệ thống 50
2.2.4.2. Mạch điều khiển rơle 51

2.2.5. Khối hiển thị LED 53
2.2.6. Khối điều khiển thiết bị từ xa 55
2.2.7. Khối giao tiếp với máy vi tính 55
3. Sơ đồ nguyên lý tổng thể hệ thống 55
4. Lưu đồ thuật toán cho phần mềm 59
4.1. Lưu đồ thuật toán của chương trình mạch điều khiển trung tâm 59
4.2. Lưu đồ thuật toán của chương trình mạch điều khiển cửa 63
4.3. Lưu đồ thuật toán của phần mềm trên máy tính 66

PHẦN 4. KẾT LUẬN 68
Phụ lục 69
Bảng mã X10 70
Chương trình cho mạch điều khiển cửa 71
Chương trình cho mạch điều khiển trung tâm 80
Mã nguồn của phần mềm trên máy tính 93
Tài liệu tham khảo

Tóm tắt luận văn

Thuật ngữ "ngôi nhà thông minh" có thể hiểu đơn giản là những ngôi nhà với những trang thiết bị tự động được thiết kế nhằm tạo ra sự an toàn và tiện lợi nhất cho người dùng. Hiện nay, rất nhiều ngôi nhà thông minh đã được đưa vào sử dụng khá thành công trên thế giới và ngay cả ở Việt Nam. Nhiều chuẩn công nghệ được sử dụng trong việc thiết kế ngôi nhà thông minh. Luận văn sẽ trình bày sơ lược về một số chuẩn phổ biến này. Luận văn sẽ trình bày kĩ các kiến thức lí thuyết và quá trình tính toán thiết kế cho hệ thống điều khiển nhỏ đã tạo ra. Các kiến thức đã được học trong môn kĩ thuật điện tử và vi xử lí là nền tảng trong quá trình thiết kế hệ thống.

Lời mở đầu

Ngày nay cùng với sự phát triển của các ngành khoa học kỹ thuật, kỹ thuật điện tử –viễn thông và công nghệ thông tin đóng vai trò quan trọng trong mọi lĩnh vực khoa học và đời sống con người. Do đó chúng ta phải nắm bắt và vận dụng nó một cách có hiệu quả nhằm góp phần vào phục vụ đời sống con người, phát triển nền khoa học kỹ thuật thế giới nói chung và kỹ thuật điện tử-viễn thông, công nghệ thông tin nói riêng.

Sự phát triển với tốc độ chóng mặt của các ngành kỹ thuật này đã biến những mơ ước của con người chúng ta trước đây thành hiện thực. Đó là mơ ước về một ngôi nhà đầy đủ tiện nghi, đáp ứng tốt nhất những nhu cầu về học tập, sinh hoạt và giải trí. Những thiết bị trong nhà có thể được tự động điều khiển bằng các bộ vi xử lí, có thể trao đổi các lệnh điều khiển, hình ảnh, âm thanh với nhau thông qua các cáp tín hiệu, đường dây điện lực hoặc không dây. Những thiết bị này sẽ được quản lý bởi một hệ thống điều khiển trung tâm. Hệ thống sẽ hiển thị tất cả các thông tin có liên quan đến việc quản lý ngôi nhà như : nhiệt độ, ánh sáng, an ninh ... Các thiết bị có thể điều khiển từ trung tâm hoặc các bảng điều khiển cầm tay hoặc nhiều phương pháp hiện đại hơn như qua mạng điện thoại, mạng Internet hoặc bằng chính giọng nói của chúng ta… Nhiều thuật ngữ mô tả những ngôi nhà được thiết kế với các ý tưởng như thế : Smart home – Ngôi nhà thông minh, Digital home –Ngôi nhà số, Home of the future –Ngôi nhà tương lai… Mặc dù đã có khá nhiều sản phẩm ứng dụng thành công, công nghệ nhà thông minh vẫn là một trong những đề tài nóng hổi đang được nghiên cứu và phát triển trên thế giới. Công nghệ nhà thông minh không nên được nghĩ đến là một công nghệ chỉ để tạo ra một ngôi nhà lộng lẫy, đầy đủ tiện nghi cho sinh hoạt, giải trí bởi hiện nay nó cũng đã có rất nhiều đóng góp cho các công việc phúc lợi xã hội như chăm sóc sức khỏe cho người già, người tàn tật…

Việc chọn lựa đề tài 'Ngôi nhà thông minh' cho luận văn tốt nghiệp xuất phát từ những ứng dụng hữu ích mà công nghệ nhà thông minh mang lại. Nội dung của đề tài gồm 3 phần chính :

- ✓ Phần 1 – Tổng quan về ngôi nhà thông minh : trình bày sơ lược các chuẩn và hệ thống được sử dụng trong công nghệ nhà thông minh
- ✓ Phần 2 – Kiến thức hỗ trợ : đưa ra các kiến thức cần thiết cho việc thiết kế mạch
- ✓ Phần 3 – Tính toán- thiết kế mạch : trình bày nguyên lý hoạt động và quá trình thiết kế mạch
- ✓ Phần 4 – Kết luận : đánh giá về sản phẩm thu được và hướng phát triển đề tài

Phần phụ lục ở cuối sẽ trình bày các mã nguồn đã lập trình cho các mạch

Nhiệm vụ của đề tài được giới hạn ở việc tạo ra một hệ thống nhỏ có khả năng quản lý và điều khiển các thiết bị trong nhà về mặt an ninh, ánh sáng và nhiệt độ. Các nhiệm vụ cụ thể sẽ được trình bày ở mục 4 phần 1 sau khi tìm hiểu các hệ thống trong ngôi nhà thông minh.

PHẦN 1. TỔNG QUAN VỀ NGÔI NHÀ THÔNG MINH

1. Khái niệm :

Ngôi nhà thông minh là ngôi nhà với các trang thiết bị được tự động hóa nhằm mục đích tạo sự tiện lợi, an toàn và vui vẻ cho cuộc sống. Công nghệ nhà thông minh cho phép tất cả các thiết bị dùng trong gia đình có thể "nói chuyện" (liên hệ, liên kết) được với nhau bằng một số giao thức hoặc chuẩn công nghiệp nào đó.

2. Các chuẩn cho công nghệ nhà thông minh :

Nhiều chuẩn khác nhau được dùng trong công nghệ nhà thông minh như INSTEON, X10, Universal Power Bus, KNX Z-wave , ZigBee. INSTEON hiện là chuẩn có nhiều ưu điểm nhất.

2.1.1. Chuẩn X10 :

2.1.1.1. Giới thiệu :

X10 là một chuẩn mở quốc tế dùng cho giao tiếp giữa các thiết bị trong công nghệ nhà thông minh. X10 . X10 được phát triển từ 1975 bởi một công ty điện tử ở Scotland nhằm mục đích điều khiển từ xa các trang thiết bị, dụng cụ trong nhà. Trên thị trường tuy có nhiều chuẩn đạt được băng thông cao hơn X10 nhưng các thiết bị X10 vẫn đang được sử dụng rộng rãi trên thế giới

2.1.1.2. Nguyên lý hoạt động :

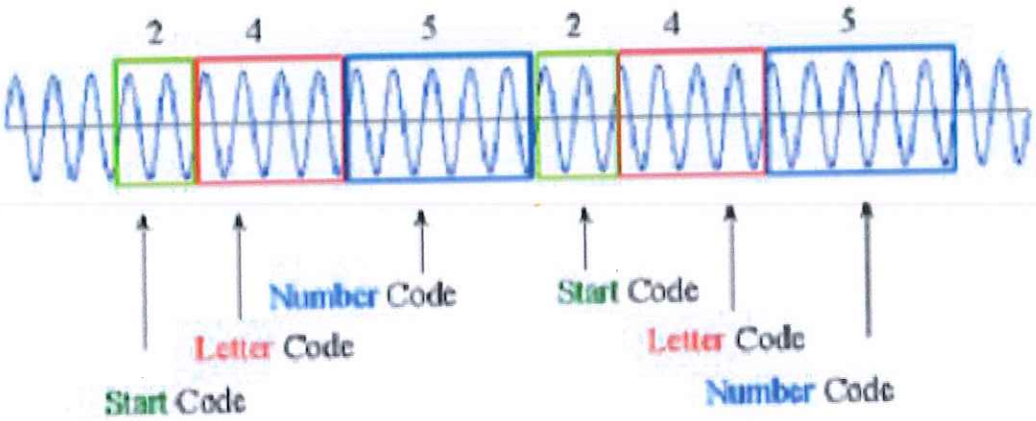
X10 sử dụng mạng điện trong nhà để gửi đi các tín hiệu số. Sóng mang được sử dụng có tần số 120Khz được phát đi tại các thời điểm qua 0 của dòng điện xoay chiều 50 hoặc 60 Hz . Tốc độ truyền khoảng 20 bit / giây

Dữ liệu số được bộ điều khiển controller phát đi bao gồm địa chỉ và lệnh đến các thiết bị được điều khiển. Việc thông báo trạng thái các thiết bị (on/off, thông số cảm biến…) theo chiều ngược lại cũng có thể được thực hiện khi sử dụng các thiết bị X10 cao cấp.

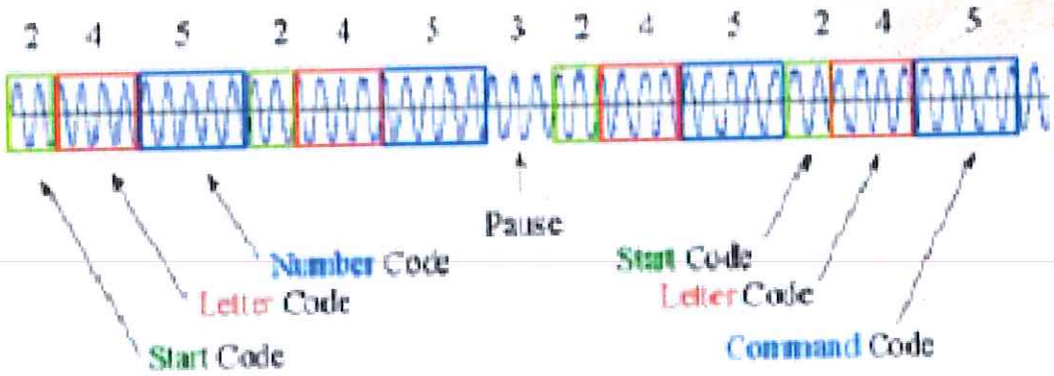
Các gói tin được phát theo chuẩn X10 chứa 4 bit "house / letter code" – mã nhà , theo sau là 4 bit "unit / number code" –mã thiết bị và cuối cùng là 4 bit "commmand code" –mã lệnh. Mã nhà được đại diện bởi các ký tự từ A đến P trong

khi mã thiết bị được đánh số từ 1 đến 16. Không có sự hạn chế nào khi sử dụng mã nhà cho thiết bị (trừ khi xem xét đến các thiết bị X10 của các nhà lân cận), vì vậy có thể có thể điều khiển 256 thiết bị cho một nhà đơn.

Cấu trúc một khung X10 :



Hình trên cho thấy 1 khung X10 được truyền lặp lại 2 lần để đảm bảo độ tin cậy của tín hiệu thu được. 'Start code' được truyền đầu tiên trong 2 chu kỳ sin với mã cố định thông báo điểm bắt đầu của khung dữ liệu.Theo sau là 4 chu kỳ sin truyền mã nhà và 5 chu kỳ truyền mã thiết bị. Sau 2 khung này, X10 sẽ chờ 6 điểm 0 của tín hiệu sin rồi mới truyền tiếp 2 khung tương tự trên nhưng lần này mã thiết bị được thay bằng mã lệnh :



Bảng mã X10 được cho ở phần phụ lục.

2.1.1.3. Bộ điều khiển (controller) :

Một bộ điều khiển đơn giản gồm các nút chọn thiết bị và các nút chức năng (như bật, tặt, điều khiển độ sáng…).

Các bộ điều khiển từ xa gồm 1 cặp thu phát dùng sóng hồng ngoại hoặc vô tuyến. Tín hiệu hồng ngoại hoặc vô tuyến sau khi được bộ thu nhận sẽ chuyển thành tín hiệu X10 phù hợp để truyền trên đường dây điện..

Một số bộ điều khiển có giao diện với PC qua cổng nối tiếp, USB để tải chương trình hoạt động từ PC. Ngoài ra , PC cũng có thể trở thành một bộ điều khiển X10 thông qua việc lập trình điều khiển các thiết bị thu phát X10.

2.1.1.4. Thiết bị được điều khiển (device module):

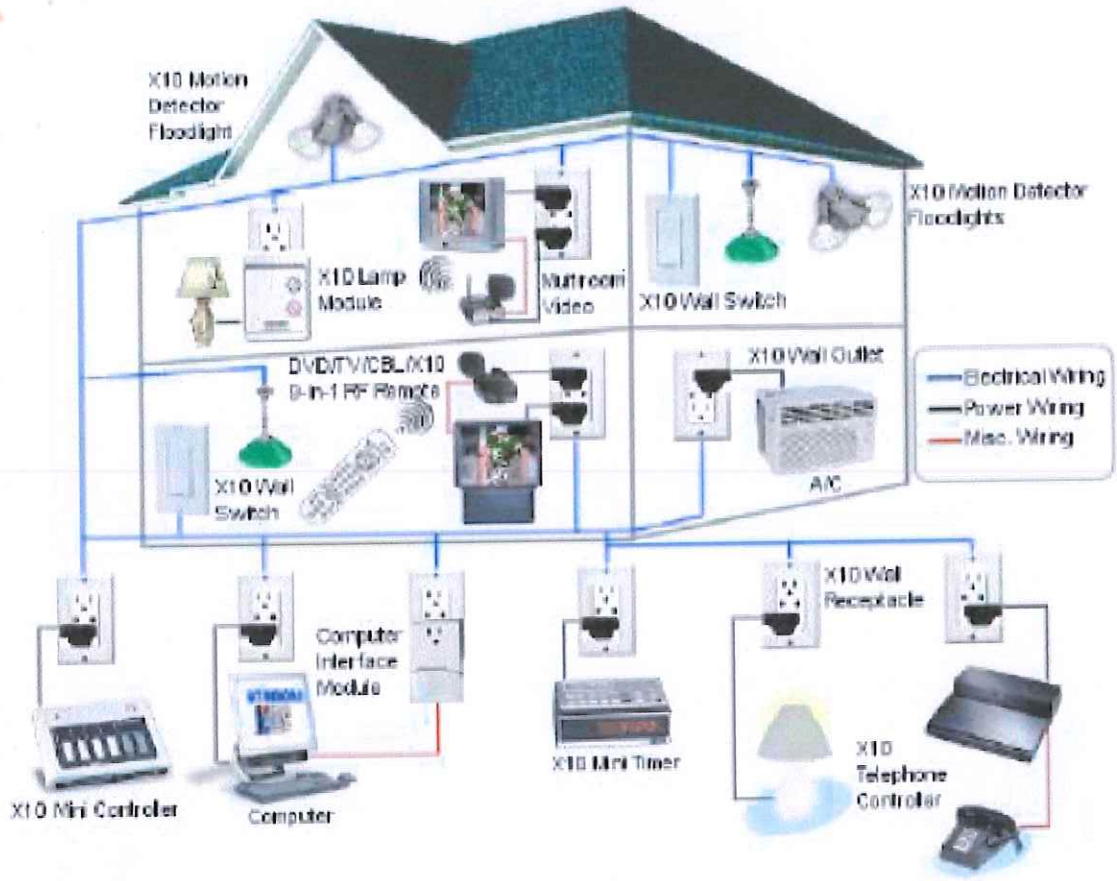
Nhiều thiết bị khác nhau với các chức năng khác nhau :

- ✓ Lamp module : điều khiển tắt / mở, điều chỉnh độ sáng tối cho các đèn sợi đốt khoảng 300W
- ✓ Appliance module : điều khiển tắt mở các đèn huỳnh quang, thiết bị điện khác như TV, các thiết bị âm thanh, … có thể dùng tắt mở đèn nhưng không chỉnh được độ sáng tối
- ✓ Wall switch module : điều khiển tắt mở , điều chỉnh độ sáng tối cho các đèn sợi đốt khoảng 500 W

Ngoài ra , còn nhiều thiết bị có chức năng đặc biệt hơn như chuyển đổi các tone điện thoại thành tín hiệu X10 giúp ta có thể điều khiển các thiết bị trong nhà từ mạng điện thoại công cộng ; các thiết bị gắn với các cảm biến ánh sáng, phát hiện chuyển động để có thể tự động điều khiển các thiết bị trong nhà hoặc ngoài sân…

2.1.1.5. Mô hình thực tế hệ thống X10 :

Mô hình sau cho thấy các thiết bị X10 khác nhau được sử dụng :



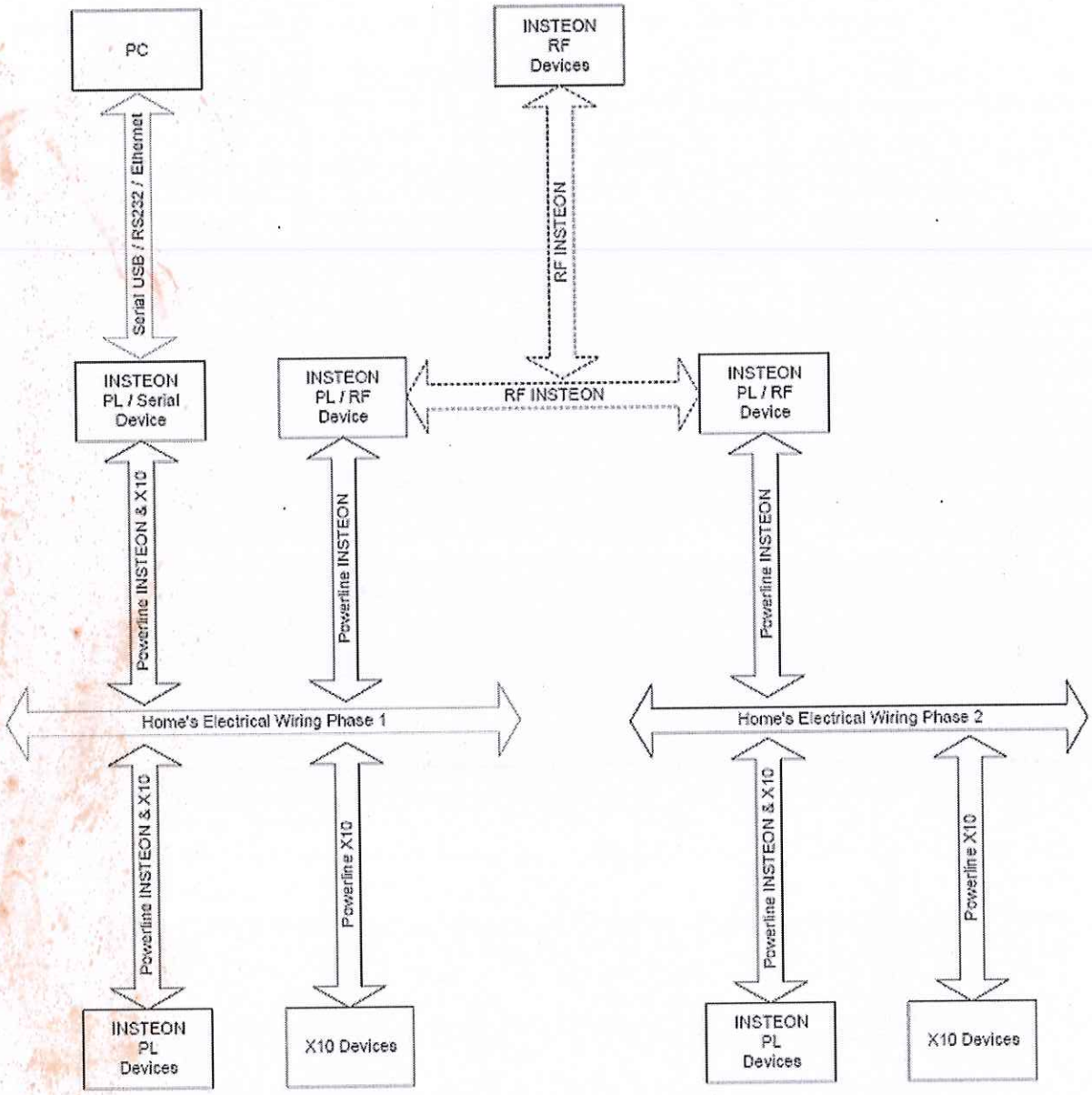
2.1.2. Chuẩn INSTEON :

2.1.2.1. Giới thiệu :

INSTEON được phát triển bởi tổ chức SmartLabs, một trong những tổ đi đầu trong công nghệ tự động hóa nhà cửa. INSTEON được phát triển từ 2001 với mục đích tạo ra một mạng điều khiển trong nhà lý tưởng, một mạng đơn giản, linh động và không đắt đỏ cho việc kết nối mọi thiết bị với nhau. Các phần mềm INSTEON đơn giản bởi các thiết bị INSTEON gửi và nhận gói tin theo cùng một cách, không đòi hỏi một bộ điều khiển mạng đặc biệt và các thuật toán định tuyến phức tạp. INSTEON được thiết kế cho các ứng dụng điều khiển nhà cửa , không phải truyền tải 1 lượng dữ liệu nào lớn. Ngoài ra , INSTEON có thể hỗ trợ các thiết bị X10 trong mạng của mình .

2.1.2.2. Nguyên lý hoạt động :

Các thiết bị INSTEON có thể giao tiếp nhau thông qua sóng vô tuyến hoặc trên đường dây điện trong nhà như hình sau :

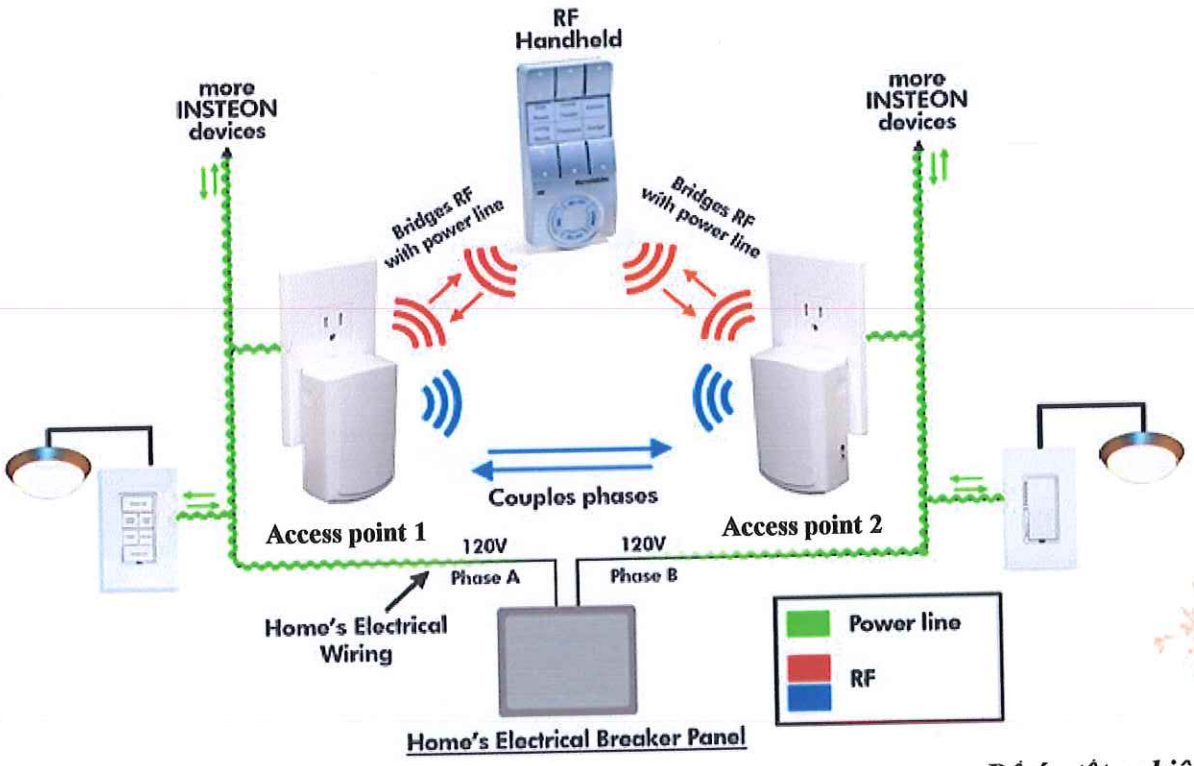


Mạng điện ở 220 VAC 3 dây ở Bắc Mỹ thường được tách ra thành 2 pha riêng biệt, mỗi pha 2 dây tại các hộp điện chính dẫn vào nhà. Vì vậy ở mỗi pha cần ít nhất một thiết bị INSTEON hỗ trợ truyền tín hiệu INSTEON trên cả 2 môi trường không khí và cáp điện lực để các thiết bị ở 2 pha có thể giao tiếp với nhau.

Các thiết bị INSTEON đều có chức năng phát , nhận và lặp tín hiệu INSTEON. Các thiết bị INSTEON có số nhận dạng riêng được cài đặt khi sản xuất, tượng trưng cho địa chỉ của thiết bị trong toàn mạng. Một gói tin INSTEON có chiều dài cố định và cũng được phát đồng bộ với các điểm 0 của tín hiệu xoay chiều 50-60 Hz. Địa chỉ nguồn và đích và mã lệnh luôn gửi kèm trong gói tin INSTEON. Mỗi địa chỉ chiếm 3 byte khiến số thiết bị điều khiển có thể lên đến con số 16,777,216. So với X10, gói tin INSTEON dài 10 byte hoặc 24 byte do chứa thêm các trường phục vụ cho quá trình truyền thông tin giữa các thiết bị. Một số thông số của INSTEON :

Giao thức	Truyền lại nếu không được báo nhận
Tốc độ truyền	2880 -13165 bit / giây
Số thiết bị hỗ trợ	16777216
Số lệnh điều khiển	65536
Tần số/ kiểu điều chế tín hiệu trên cáp điện	131,65 Khz / BPSK
Tần số/ kiểu điều chế tín hiệu vô tuyến	904 Mhz / FSK

2.1.2.3. Mô hình thực tế hệ thống INSTEON :



Thiết bị điều khiển hệ thống thông qua sóng vô tuyến (RF handheld) sẽ gửi lệnh đến điểm truy cập (Access point), tín hiệu RF sẽ được chuyển đổi thành tín hiệu INSTEON truyền trên cáp điện. Các điểm truy cập còn có chức năng trao đổi tín hiệu giữa 2 pha với nhau. Điều khiển không dây có thể được thay thế bằng các bảng điều khiển có dây với giá thành rẻ hơn.

Ngoài X10 và INSTEON, có thể kể đến 1 số chuẩn khác như :

- ✓ Universal Power Bus (UPB) : sử dụng cáp điện lực
- ✓ KNX : sử dụng cho các tòa nhà thông minh hỗ trợ cáp xoắn, cáp điện lực, vô tuyến, hồng ngoại và Ethernet
- ✓ Z-wave , ZigBee : giao tiếp không dây

3. Các hệ thống trong ngôi nhà thông minh :

Dựa trên các chuẩn công nghiệp, một ngôi nhà thông minh có thể gồm các hệ thống sau :

- ✓ Hệ thống ánh sáng, nhiệt : được điều khiển từ xa hoặc nhân công đèn huỳnh quang tắt/mở, các đèn sợi đốt tắt/mở, sáng/tối.
- ✓ Hệ thống tưới nước khuôn viên: được điều khiển nhiều vùng bằng bộ điều khiển từ xa qua các sensor cảm biến hoặc thông qua thời gian biểu được lập trình trước trên máy tính.
- ✓ Hệ thống điều khiển rèm cửa tự động: Nhân công hoặc tự động điều khiển.
- ✓ Hệ thống tự động chiếu sáng cầu thang và phòng vệ sinh: Tự động bật sáng khi có người và tắt sau vài giây.Ban ngày đèn không sáng.
- ✓ Hệ thống phục vụ giải trí: âm thanh và hình ảnh được bật tắt và phát ở các phòng khác nhau.
- ✓ Hệ thống an ninh :
    - o Các camera với chức năng ghi hình liên tục hoặc ghi hình truyền qua mạng

- Các cảm biến phát hiện chuyển động, phát hiện tiếp xúc với cửa chính / cửa sổ, phát hiện các cửa kính bị đập vỡ … nhằm chống xâm nhập từ bên ngoài.
- Các khóa cửa sử dụng mật mã hoặc nhận dạng qua vân tay
- Có khả năng phát hiện cháy, rò rỉ gas… và phát báo động kịp thời

Các phương thức điều khiển cho ngôi nhà thông minh bao gồm :

- ✓ Nhân công
- ✓ Từ xa có dây/ không dây
- ✓ Điều khiển bằng giọng nói
- ✓ Điều khiển thông qua mạng điện thoại công cộng, điện thoại di động, internet

4. <u>Giới hạn nhiệm vụ đồ án:</u>

Thiết kế mạch đảm bảo an ninh :

Sử dụng vi điều khiển, bàn phím, màn hình LCD nhỏ. Password phải được nhập đúng thì cửa mới mở. Khi vào thì ấn 1 phím chức năng, cửa sẽ tự động đóng lại. Có phím chức năng "chuông cửa", khi ấn hệ thống loa máy tính sẽ phát chuông

Thiết kế mạch cảm biến ánh sáng, nhiệt độ, báo cháy và điều khiển thiết bị :

Sử dụng vi điều khiển, máy tính, 2 cảm biến ánh sáng , 2 cảm biến nhiệt gắn ở 2 phòng khác nhau . Hoạt động của mạch như sau :

- ✓ Độ sáng, nhiệt độ trong phòng sẽ được vi điều khiển nhận biết thông qua các cảm biến. Tùy theo độ sáng và nhiệt độ, vi điều khiển sẽ tự động bật tắt các thiết bị đèn quạt gắn ở từng phòng
- ✓ Khi nhiệt độ tăng cao quá mức, hệ thống loa (của máy tính) sẽ phát âm thanh báo động.
- ✓ Máy tính có thể hiển thị trạng thái các thiết bị trong nhà, nhiệt độ của từng phòng, có thể điều khiển các thiết bị và cài đặt các thông số về ánh sáng, nhiệt độ trên giao diện phần mềm của máy tính .
- ✓ Ngoài ra, các thiết bị trong phòng còn có thể điều khiển từ xa thông qua cặp IC SC9148, SC9149.

PHẦN 2 : KIẾN THỨC HỖ TRỢ

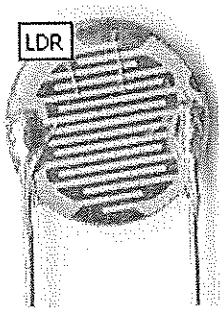
1. Cảm biến :

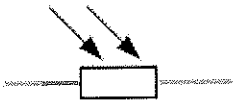
Cảm biến là một thiết bị có khả năng biến đổi dạng năng lượng này thành dạng năng lượng khác dùng cho các mục đích khác nhau đặc biệt là trong đo lường. Một số cảm biến có thể chỉ thị kết quả đo trực tiếp như nhiệt kế thủy ngân nhưng một số loại phải đi kèm các thiết bị chỉ thị khác như cặp nhiệt điện.

1.1. Cảm biến ánh sáng – Quang trở :

1.1.1. Mô tả hoạt động :

Quang trở là loại linh kiện có điện trở thay đổi theo cường độ ánh sáng chiếu vào. Thông thường chúng được chế tạo từ loại vật liệu có điện trở suất phụ thuộc nhiều vào ánh sáng như : Cadmium Sulfid (CdS), Cadmium Selenide (CdSe) hoặc Cadmium Telluride (CdTe). Vật liệu bán dẫn này được chế tạo thành hình zigzac, mạ kim loại ở hai đầu để gắn điện cực, đặt trong vỏ nhựa hoặc vỏ kim loại có « cửa sổ » trong suốt nhận ánh sáng chiếu vào. Hình dạng và kí hiệu quang trở như sau :





Khi có ánh sáng có với cường độ và bước sóng thích hợp chiếu vào , nhờ trao đổi năng lượng với các photon, một số điện tử và lỗ trống được giải phóng ra. Nếu quá trình hấp thụ ánh sáng xảy ra ở một loại bán dẫn tạp chất nào đó, năng lượng của Photon sẽ làm ion hóa nguyên tử tạp chất, giải phóng ra loại hạt dẫn tương ứng. Chẳng hạn ở bán dẫn loại N, hiện tượng ion hóa các tạp chất donor sẽ tạo sự biến thiên nồng độ điện tử. Tương tự, ở bán dẫn loại P, quá trình ion hóa tạp chất acceptor gây ra biến thiên nồng độ lỗ trống.

Quá trình làm thay đổi nồng độ hạt dẫn do tác dụng của ánh sáng không phải là tức thời. Lượng biến thiên nồng độ hạt dẫn hoặc lỗ trống sẽ tăng theo hàm mũ từ thời điểm ánh sáng bắt đầu kích thích và giảm theo hàm mũ khi tắt nguồn sáng chứ không đột biến ngay về zero

1.1.2. Các đặc tuyến của quang trở :

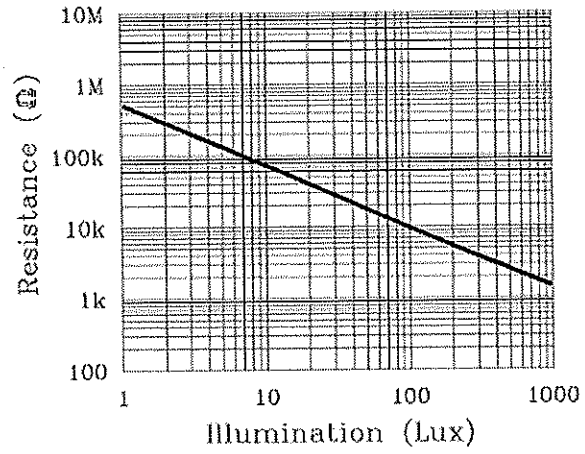
Đặc tuyến điện trở :

Quang thông ϕ có đơn vị Lumen (lm) trong hệ SI được dùng để đo công suất ánh sáng nhận được. Độ rọi là quang thông tổng cộng trên một đơn vị diện tích. Độ rọi có đơn vị là lux (lx) hoặc lm/m^2.

Điện trở của quang trở sẽ tỉ lệ tuyến tính với quang thông trên một đơn vị diện tích.

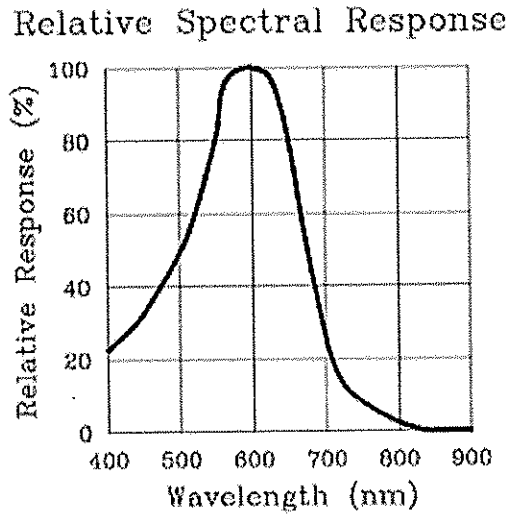
$$R = R_0 * \phi^{-k}$$ R_0 , k là hằng số (k <1)

Đặc tuyến điện trở của một quang trở điển hình được cho trên hình sau :



Đặc tuyến phổ :

Độ nhậy quang biểu thị trị số dòng điện sáng ứng với quang thông là đơn vị. Độ nhậy quang tương đối là tỉ số giá trị độ nhạy quang so với giá trị cực đại của nó. Đặc tuyến phổ của một quang trở CdS điển hình được cho trên hình sau :



Trục tung biểu thị giá trị tương đối của độ nhạy. Tùy theo mỗi vật liệu chế tạo mà từng loại quang trở nhạy cảm với một dãy bước sóng khác nhau. Hình trên cho thấy loại quang trở được chế tạo cho vùng ánh sáng nhìn thấy được.

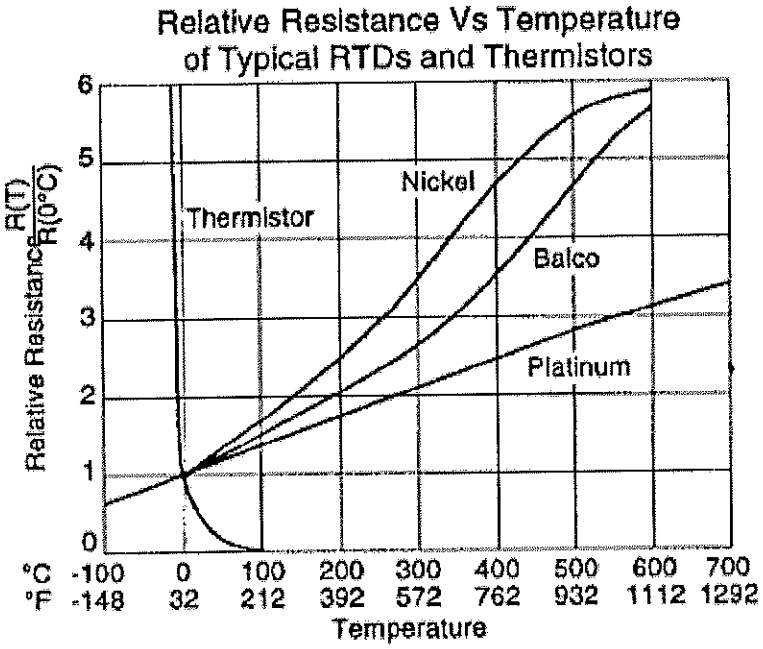
1.2. Cảm biến nhiệt độ :

1.2.1. Nhiệt điện trở kim loại :

1.2.1.1. Mô tả hoạt động :

Nhiệt điện trở kim loại (Resistance thermometer / Resistance temperature detector-RTD) là 1 loại cảm biến nhiệt lợi dụng đặc tính thay đổi điện trở biết được của một số vật liệu đối với nhiệt độ. Chúng thường được chế tạo bằng Platin (do có sự ổn định và điện trở thay đổi gần như tuyến tính với nhiệt độ) và dần thay thế các cặp nhiệt điện trong các ứng dụng công nghiệp dưới 600 °C. Cho đến hiện nay, đa số các RTD dùng trong công nghiệp có giá trị điện trở 100 ohm ở 0 °C, và hệ số nhiệt điện trở alpha 0.00385. Alpha được định nghĩa là độ thay đổi điện trở trung bình trên 1 °C trong dải từ 0-100°C chia cho điện trở tại 0°C.

Sơ đồ sau so sánh đặc tuyến điện trở -nhiệt độ của các loại nhiệt điện trở kim loại lẫn bán dẫn:



Qua sơ đồ, có thể thấy RTD là cảm biến có điện trở tuyến tính nhất với nhiệt độ.

Đối với RTD, sử dụng phương trình Callendar-Van Dusen để tính nhiệt độ hoặc điện trở :

$$R_T = R_0 (1+AT+BT^2 -100CT^3 + CT^4) \ (-200\ ^oC \le T < 0\ ^oC)$$
$$R_T = R_0 (1 + AT + BT^2) \ (0^oC \le T < 850\ ^oC)$$

R_T, R_0 : điện trở ở T °C , 0 °C

A, B, C là các tham số được rút ra qua các phép đo thực tế điện trở của RTD ở các nhiệt độ 0 °C, 100 °C, 260 °C.

Giá trị điển hình của A, B, C đối với RTD Platin có alpha = 0.00385 như sau :

$A = 3.9083 \times 10^{-3}\ ^oC^{-1}$

$B = -5.775 \times 10^{-7}\ ^oC^{-2}$

$C = -4.183 \times 10^{-12}\ ^oC^{-4}$

B, C xấp xỉ 0 nên ta có thể thấy điện trở sẽ là hàm tuyến tính của nhiệt độ.

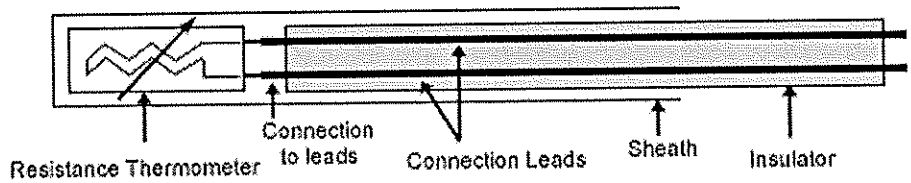
Ưu điểm :

- ✓ Độ chính xác cao : RTD thuộc lớp A(cao cấp) có độ chính xác cho bởi phương trình sau : $\Delta T = \pm(0.15 + 0.002\,|t|)$ $-200\,^{o}C < t < 650\,^{o}C$
- ✓ Dải hoạt động rộng : $-200\,^{o}C < t < 850\,^{o}C$ (RTD lớp B)
- ✓ Độ trôi nhiệt độ thấp

Giới hạn :

- ✓ Nhiệt độ đo tối đa bị hạn chế do khoảng ở 660 °C Platin sẽ lẫn tạp với các phân tử kim loại của vỏ.
- ✓ So với nhiệt điện trở bán dẫn, RTD ít nhạy với các sự thay đổi nhỏ của nhiệt độ và có thời gian đáp ứng chậm hơn.

Hình dạng cấu tạo :



Shealth : vỏ bảo vệ

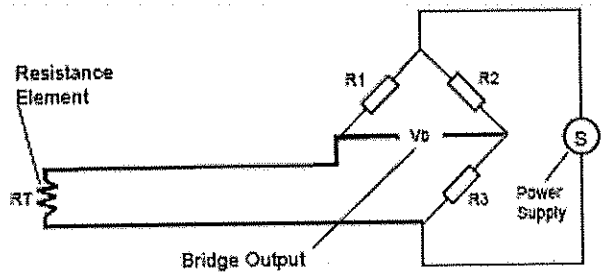
Insulator : chất cách điện dùng thủy tinh hoặc gốm

Connection leads : dây dẫn ra ngoài

Connection to leads : mối nối

1.2.1.2. <u>Sơ đồ đo :</u>

Các nhiệt điện trở chỉ cần một nguồn công suất nhỏ để hoạt động. Điện trở sẽ thay đổi tuyến tính với nhiệt độ. Hình sau là sơ đồ dùng cầu đo đo điện trở R_T từ đó tính được nhiệt độ thông qua phương trình Callendar:



1.2.2. Nhiệt điện trở bán dẫn :

1.2.2.1. Mô tả hoạt động :

Nhiệt điện trở bán dẫn (thermistor) được xem là cảm biến nhiệt có điện trở thay đổi theo nhiệt độ. Nhiệt điện trở có độ nhạy cao nhưng dải hoạt động thấp hơn so với nhiệt điện trở kim loại.

Nhiệt điện trở bán dẫn được chia làm 2 loại :

- ✓ Nhiệt điện trở có hệ số nhiệt dương (positive temperature cofficient – PTC) : điện trở tăng khi nhiệt độ tăng
- ✓ Nhiệt điện trở có hệ số nhiệt âm (negative temperature cofficient – NTC) điện trở tăng khi nhiệt độ giảm

Nhiệt điện trở bán dẫn được chế tạo từ gốm hoặc polyme

Trong các ứng dụng đo nhiệt độ, ta thường dùng NTC. Mối quan hệ giữa nhiệt độ và điện trở của các NTC được cho theo phương trình sau :

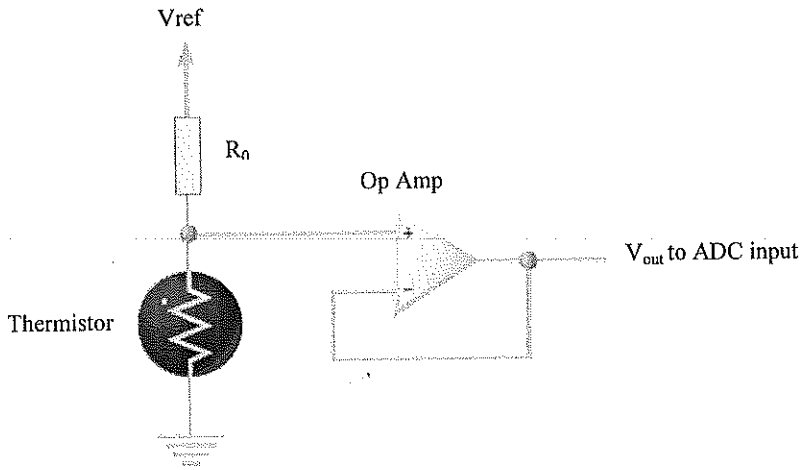
$$1/t=a+b(\text{Ln } R_t/R_{25})+c(\text{Ln } R_t/R_{25})^2+d(\text{Ln } R_t/R_{25})^3$$

R_t , R_{25} : điện trở của cảm biến tại t °C và 25 °C

a, b, c, d : các tham số phụ thuộc vật liệu chế tạo

a, b, c, d , R_{25} : được cho bởi nhà sản xuất

1.2.2.2. Sơ đồ đo :



Điện áp rơi trên NTC sẽ được đọc bởi ADC thông qua mạch lặp điện áp bằng Op Amp. Từ đó có thể tính được điện trở NTC thông qua các giá trị Vref, R_0 đã biết

1.2.3. IC chuyên dụng :

Một số IC được chế tạo cho mục đích đo nhiệt độ. IC nhận tín hiệu nhiệt độ và chuyển thành tín hiệu điện áp hoặc dòng điện tỉ lệ tuyến tính với nhiệt độ.

Một số IC phổ biến trong ứng dụng đo nhiệt độ như :

AD590 :

- ✓ Ngõ ra là dòng điện
- ✓ Sai số : có thể tối ưu độ chính xác bằng các mạch xén. Một sai số chỉ ± 0.2 ^oC trong dải từ 25 – 75 ^{o} C hoàn toàn có thể tạo ra được.
- ✓ Độ nhạy : 1 μA / ^oK
- ✓ Nguồn cung cấp : 4 V -30 V
- ✓ Dải nhiệt độ có thể đo : -55 ^oC – 150 ^oC

LM 135, LM 335 :

- ✓ Ngõ ra là điện áp, cần cân chỉnh trước ở 1 nhiệt độ nào đó
- ✓ Sai số : $\pm 0.5^oC$ (LM 135), $\pm 1^oC$ (LM 335) trong dải nhiệt độ đo
- ✓ Độ nhạy : 10 mV / ^oK
- ✓ Dải nhiệt độ có thể đo : -55 ^oC – 150 ^oC (LM 135) , -40 ^oC – 100 ^oC (LM 335)

LM 35 :

- ✓ Ngõ ra là điện áp
- ✓ Sai số : ± 0.4 ^oC (25 ^oC) , $\pm$ 0.8^oC ở (Tmax , T min)
- ✓ Độ nhạy : 10 mV / ^oC
- ✓ Dải nhiệt độ có thể đo : -50 ^oC – 150 ^oC

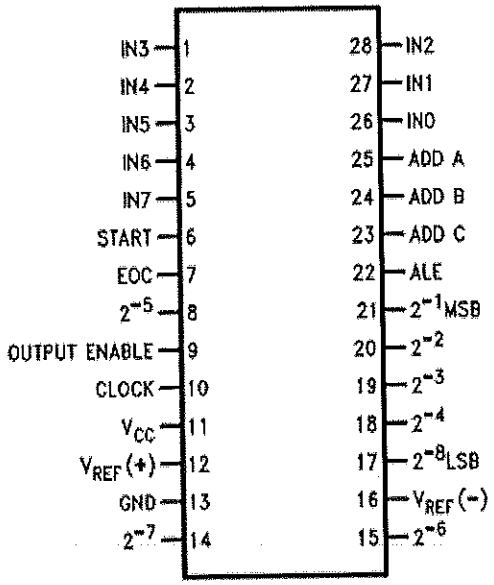
2. IC chuyển đổi số - tương tự ADC 0809 :

2.1. Giới thiệu IC:

ADC 0809 hoạt động theo nguyên tắc xấp xỉ liên tiếp. Điểm đặc biệt của nó so với ADC 0804 là nó có tới 8 ngõ vào. ADC 0809 không cần chỉnh không hoặc toàn thang. Việc giao tiếp với vi xử lí dễ dàng bởi ADC 0809 có chân chốt địa chỉ và các chân dữ liệu ra TTL 3 trạng thái.

2.2. Đặc điểm , sơ đồ chân và chức năng :

- ✓ Độ phân giải 8 bit
- ✓ Sai số tối đa khi chưa hiệu chỉnh : ± 1 LSB
- ✓ Thời gian chuyển đổi : 100 μ s
- ✓ Nguồn cung cấp : 5V
- ✓ Điện áp ngõ vào : 0 – 5 V
- ✓ Tần số xung clock : 10 Khz – 1280 Khz



Ý nghĩa các chân :

IN0 đến IN7 : 8 ngõ vào tương tự

A, B, C : địa chỉ cho các ngõ vào

2^{-1} đến 2^{-8} : chân dữ liệu ra

ALE : chân chốt địa chỉ

START : tạo phép biến đổi mới

CLK : xung clock

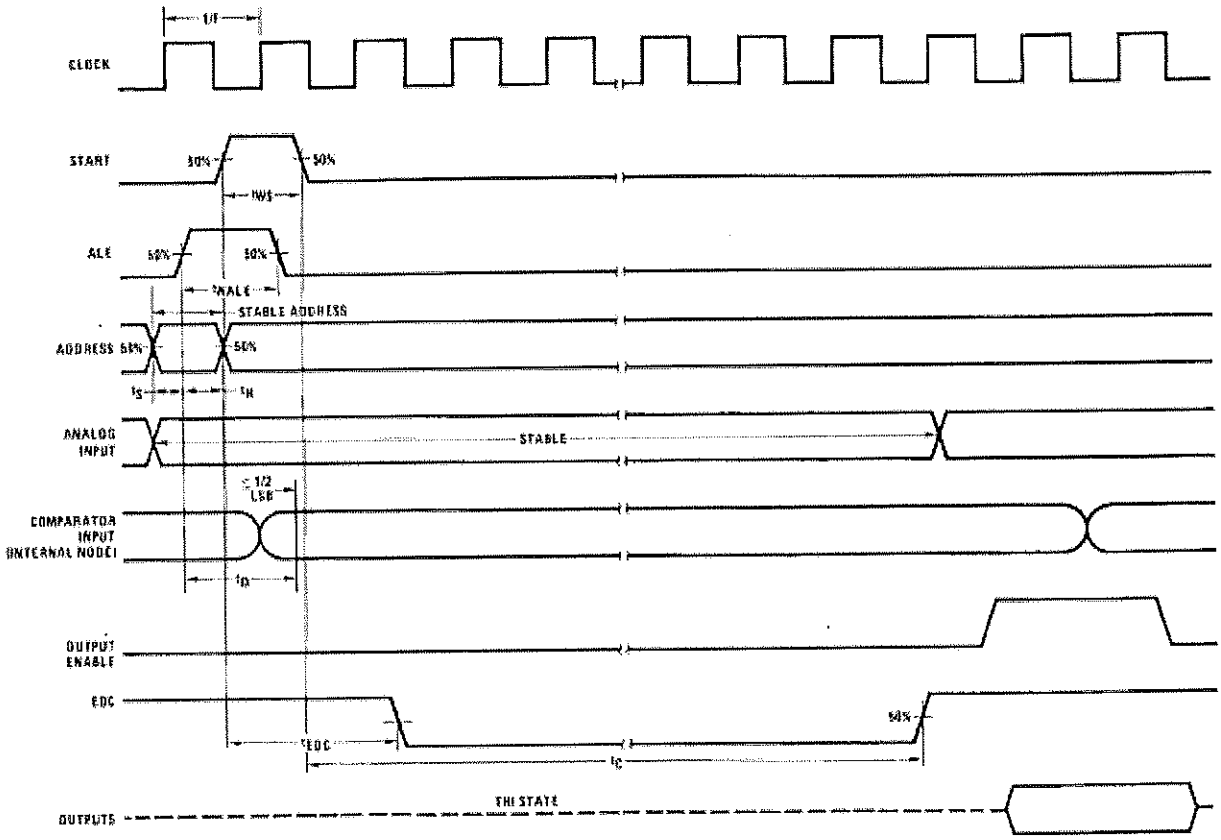
REF (+) : chân điện áp chuẩn +

REF (-) : chân điện áp chuẩn –

VCC, GND : châm nguồn và đất

2.3. Nguyên tắc hoạt động :

Hoạt động của ADC sẽ được giải thích bởi giản đồ thời gian trang sau :



Sau khi đã chốt địa chỉ thích hợp, xung START được kích thích, bộ chuyển đổi bắt đầu hoạt động ở cạnh xuống của xung START, ngõ ra EOC sẽ xuống mức thấp sau khoảng 8 xung clock tính từ cạnh xuống này. Sau khi chuyển đổi kết thúc, EOC sẽ trở lại mức cao. Dữ liệu có thể được đọc vào vi xử lí sau khi đặt OE lên mức cao.

3. Vi điều khiển 89C51

3.1. Giới thiệu IC :

Do họ MCS-51™ đã trở thành chuẩn công nghiệp nên có rất nhiều hãng sản xuất ra nó, điển hình là ATMEL Corporation. Hãng này đã kết hợp rất nhiều tính năng dựa trên nền tảng kỹ thuật của mình để tạo ra các vi điều khiển tương thích với MCS-51™ nhưng mạnh mẽ hơn.

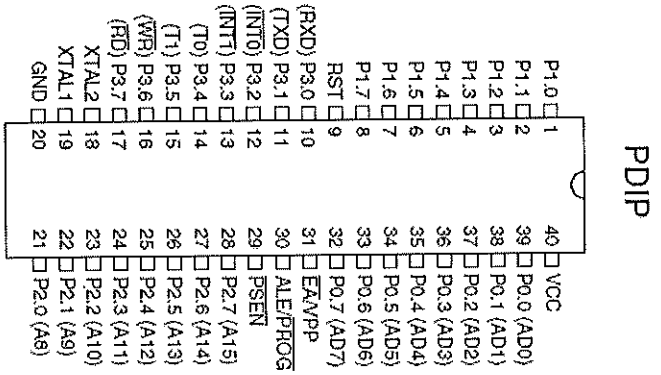
AT89C51 là một vi điều khiển mạnh (có công suất lớn), cung cấp một sự linh động cao và giải pháp về giá cả đối với nhiều ứng dụng vi điều khiển.

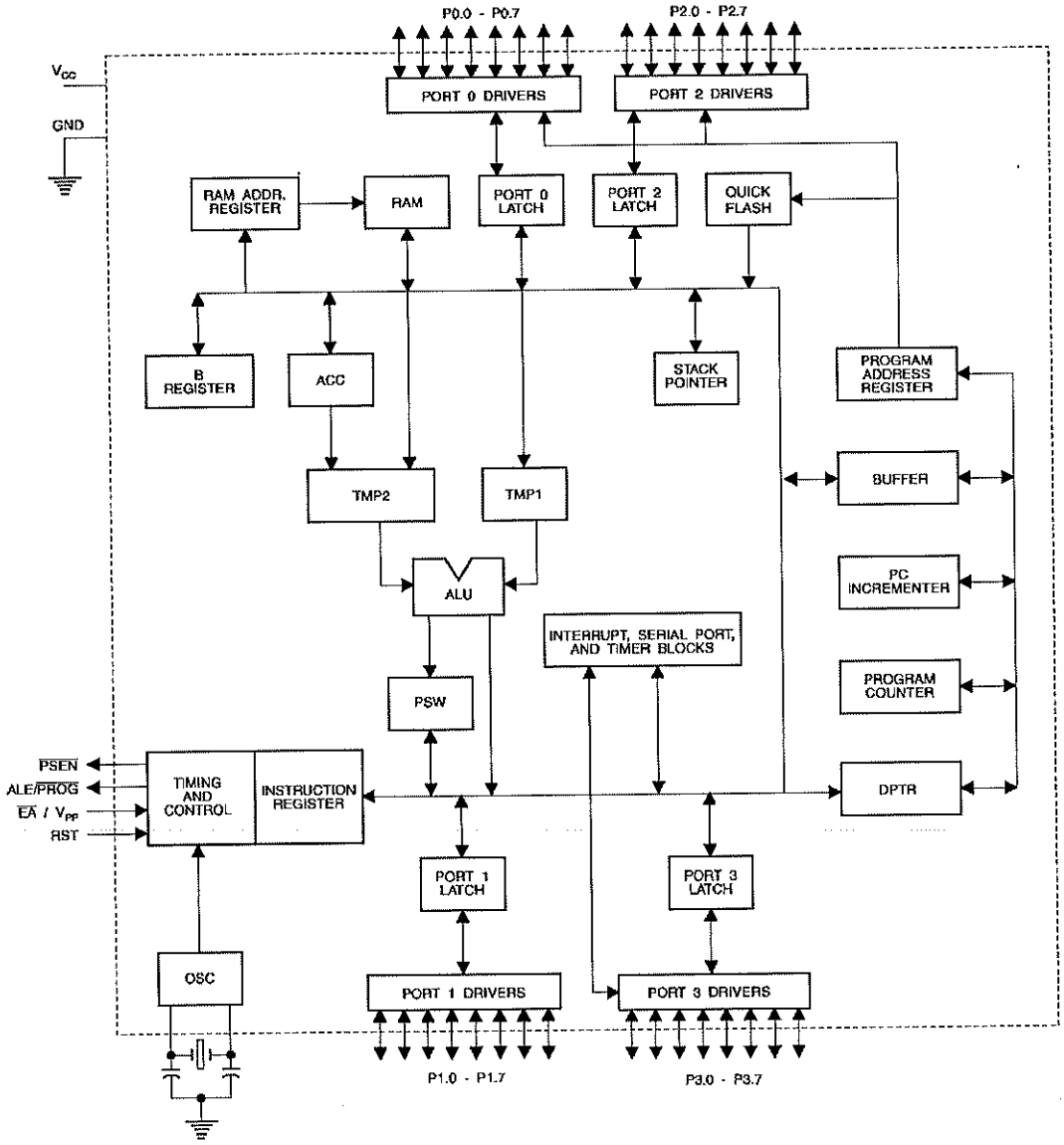
Các đặc điểm chủ yếu của AT89C51 :

- ✓ 4Kbyte bộ nhớ chương trình
- ✓ 128 byte RAM
- ✓ 32 đường I/O lập trình được (4 port).
- ✓ 2 timer/counter 16-bit.
- ✓ Tần số hoạt động : 0 Hz đến 24 MHz.
- ✓ Giao tiếp nối tiếp.
- ✓ 64 KB vùng nhớ mã ngoài.
- ✓ 64 KB vùng nhớ dữ liệu ngoài.
- ✓ Bộ xử lý bit (thao tác trên các bit riêng rẽ).
- ✓ 210 vị trí nhớ có thể định vị bit.
- ✓ Độ bền : 1000 lần ghi/xóa.
- ✓ 6 nguồn ngắt.
- ✓ 4 µs cho hoạt động nhân hoặc chia.
- ✓ Chế độ hạ nguồn và chế độ nghỉ tiêu tốn công suất thấp.

3.2. Sơ đồ chân và sơ đồ các khối bên trong :

Kiểu đóng gói IC thường sử dụng là PDIP – Plastic Dual Inline Package.





3.3. Chức năng các chân :

Vcc : chân cung cấp nguồn.

GND : chân nối đất.

Port 0 : là một port I/O 8-bit hai chiều, cực máng hở. Khi ở chế độ xuất, mỗi chân port có thể rút dòng từ 8 đầu vào TTL. Nếu ghi mức 1 ra các chân port 0 thì các chân này có thể dùng như các đầu vào trở kháng cao. Port 0 cần nối các điện trở kéo lên bên ngoài khi cần xuất mức 1.

Port 0 cũng có thể được cấu hình thành một bus đa hợp giữa địa chỉ thấp và dữ liệu khi truy cập chương trình hay dữ liệu ngoài. Trong chế độ này, port 0 có các điện trở kéo lên bên trong.

Port 1 : là một port I/O 8-bit hai chiều có các điện trở kéo lên bên trong. Đầu ra port 1 có thể lái 4 đầu vào TTL. Khi viết các mức 1 ra các chân port thì chúng được kéo lên do có điện trở nội và có thể dùng làm đầu vào..

Port 2 : là một port I/O có các đặc điểm tương tự port 1. Khi giao tiếp với bộ nhớ chương trình và bộ nhớ dữ liệu bên ngoài, port 2 đóng vai trò là byte địa chỉ cao

Port 3 : là một port I/O có các đặc điểm tương tự port 1,2 . Port 3 có thể có nhiều chức năng khác được cho trong bảng sau

BIT	TÊN	ĐỊA CHỈ BIT	CHỨC NĂNG
P3.0	RxD	B0H	Nhận dữ liệu cho port nối tiếp
P3.1	TxD	B1H	Truyền dữ liệu cho port nối tiếp
P3.2	INT0\	B2H	Ngắt ngoài 0
P3.3	INT1\	B3H	Ngắt ngoài 1
P3.4	T0	B4H	Ngõ vào từ bên ngoài cho timer/counter 0
P3.5	T1	B5H	Ngõ vào từ bên ngoài cho timer/counter 1
P3.6	WR\	B6H	Xung ghi bộ nhớ dữ liệu ngoài
P3.7	RD\	B7H	Xung đọc bộ nhớ dữ liệu ngoài

RST : là ngõ vào Reset. Khi ngõ này được đưa lên cao trong hai chu kì máy, các thanh ghi bên trong AT89C51 được tải những giá trị thích hợp để khởi động hệ thống.

ALE/PROG\ (Address Latch Enable) : ALE là xung xuất cho phép chốt byte địa chỉ thấp khi truy cập bộ nhớ ngoài. Chân này còn là ngõ vào của xung lập trình (**PROG**) khi lập trình Flash.

Trong hoạt động bình thường, ALE được phát xung với tần số 1/6 tần số của mạch dao động trên chip và có thể được dùng như xung thời gian chuẩn bên ngoài.

Có thể hủy bỏ chức năng của ALE bằng cách set bit 0 của vùng thanh ghi chức năng đặc biệt có địa chỉ 8EH. Một khi bit này được set, ALE chỉ tích cực khi có lệnh MOVX hoặc MOVC. Nếu không có các lệnh này thì ALE ở mức cao. Việc set bit 0 của thanh ghi ở vị trí 8EH không làm ảnh hưởng đến vi điều khiển khi truy cập bộ nhớ ngoài.

PSEN\ (Program Store Enable) : PSEN là xung báo hiệu việc đọc bộ nhớ chương trình ngoài.. Khi thi hành chương trình trong RAM nội, PSEN sẽ ở mức thụ động (mức cao).

EA\ / VPP (External Access) : EA\ là ngõ vào để cho phép truy xuất bộ nhớ chương trình từ bên ngoài khi được nối với GND. Khi EA\ được kéo lên nguồn VCC, chương trình sẽ được thực thi trong ROM nội.

XTAL1 : Đầu vào của bộ khuếch đại dao động đảo và cũng là đầu vào mạch tạo xung clock bên trong.

XTAL2 : Đầu ra của bộ khuếch đại dao động đảo.

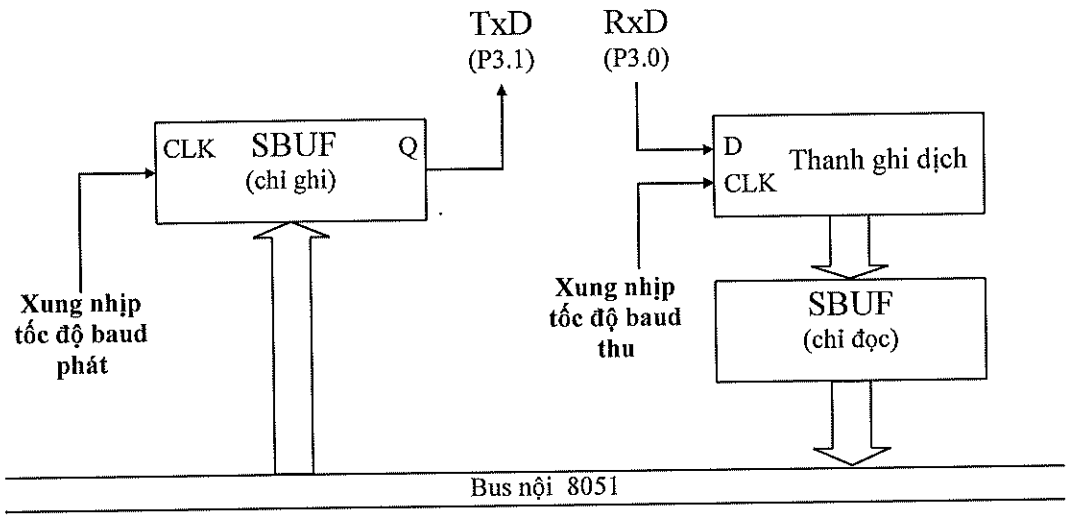
3.4. Giới thiệu port nối tiếp và thanh ghi điều khiển SCON :

AT8951 có một port nối tiếp trong chip có thể hoạt động ở nhiều chế độ trên một dãi tần số rộng. Chức năng chủ yếu của port nối tiếp là thực hiện chuyển đổi song song sang nối tiếp khi truyền dữ liệu, và chuyển đổi nối tiếp sang song song khi nhận dữ liệu. Truy xuất phần cứng đến port nối tiếp qua các chân TxD (P3.1) và RxD (P3.0).

Port nối tiếp cho hoạt động song công (full duplex: thu và phát đồng thời), và đệm lúc thu (receiver buffering), cho phép một ký tự sẽ được thu và được giữ trong khi ký tự thứ hai được nhận. Nếu CPU đọc ký tự thứ nhất trước khi ký tự thứ hai được thu được thu đầy đủ thì dữ liệu sẽ không bị mất.

Hai thanh ghi chức năng đặc biệt cho phép phần mềm truy xuất đến port nối tiếp là: SBUF và SCON. Bộ đệm port nối tiếp (SBUF) ở địa chỉ 99H thật sự là hai bộ đệm. Ghi vào SBUF để nạp dữ liệu sẽ phát, và đọc SBUF để truy xuất dữ liệu thu được. Đây là hai thanh ghi riêng biệt .

Sơ đồ khối Port nối tiếp :



Thanh ghi điều khiển port nối tiếp (SCON) ở địa chỉ 98H là thanh ghi có địa chỉ bit chứa các bit trạng thái và các bit điều khiển. Các bit điều khiển đặt chế độ hoạt động cho port nối tiếp, và các bit trạng thái báo kết thúc việc phát hoặc thu ký tự. Các bit trạng thái có thể được kiểm tra bằng phần mềm hoặc có thể được lập trình để tạo ngắt.

Tần số làm việc của port nối tiếp, còn gọi là tốc độ baud có thể cố định (lấy từ bộ dao động trên chip) hoặc thay đổi. Nếu sử dụng tốc độ baud thay đổi, Timer 1 sẽ cung cấp xung nhịp tốc độ baud và phải được lập trình.

Chế độ hoạt động của port nối tiếp được đặt bằng cách ghi từ điều khiển vào thanh ghi chế độ port nối tiếp (SCON) ở địa chỉ 98H.

Bảng 3: Thanh ghi SCON

Bit	Ký hiệu	Địa chỉ	Mô tả
SCON.7	SM0	9FH	Bit 0 của chế độ port nối tiếp.
SCON.6	SM1	9EH	Bit 1 của chế độ port nối tiếp.
SCON.5	SM2	9DH	Bit 2 của chế độ port nối tiếp. Cho phép truyền thông đa xử lý trong các chế độ 2 và 3; RI sẽ không bị tác động nếu bit thứ 9 thu được là 0.
SCON.4	REN	9CH	Cho phép thu, phải được đặt lên 1 để thu (nhận) các ký tự.
SCON.3	TB8	9BH	Bit thứ 9 được phát trong các chế độ 2 và 3; được đặt và xóa bằng phần mềm.
SCON.2	RB8	9AH	Bit thứ 9 thu được.
SCON.1	TI	99H	Cờ ngắt phát. Đặt lên 1 khi kết thúc phát ký tự; được xóa bằng phần mềm.
SCON.0	RI	98H	Cờ ngắt thu. Đặt lên 1 khi kết thúc thu ký tự; được xóa bằng phần mềm.

3.5. Các chế độ hoạt động của port nối tiếp :

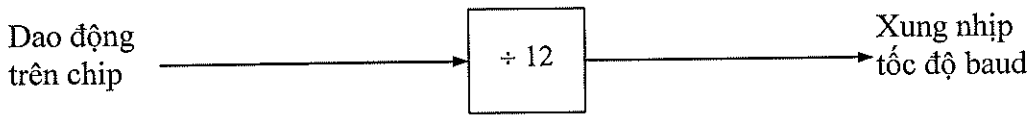
Port nối tiếp có 4 chế độ hoạt động, trong đó ba chế độ cho phép truyền thông bất đồng bộ, với mỗi ký tự được thu hoặc phát đều bắt đầu bằng một bit Start và kết thúc bằng 1 bit Stop. Ở chế độ thứ tư, port nối tiếp hoạt động như một thanh ghi dịch đơn giản.

Các chế độ của Port nối tiếp được cấu hình như sau :

SM0	SM1	Chế độ	Mô tả	Tốc độ baud
0	0	0	Thanh ghi dịch	Cố định (tần số dao động/12).
0	1	1	UART 8 bit	Thay đổi (thiết lập bằng bộ định thời).
1	0	2	UART 9 bit	Cố định (tần số dao động/32 hoặc 64).
1	1	3	UART 9 bit	Thay đổi (thiết lập bằng bộ định thời).

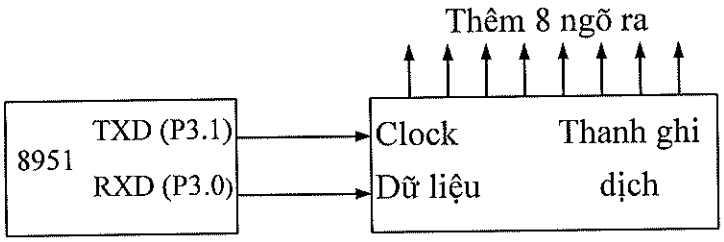
3.5.1. Chế độ 0 –Thanh ghi dịch 8 bit :

Dữ liệu nối tiếp vào và ra qua RxD, còn TxD xuất xung nhịp dịch. 8 bit được phát hoặc thu với bit đầu tiên là LSB. Tốc độ baud cố định ở 1/12 tần số dao động trên chip.



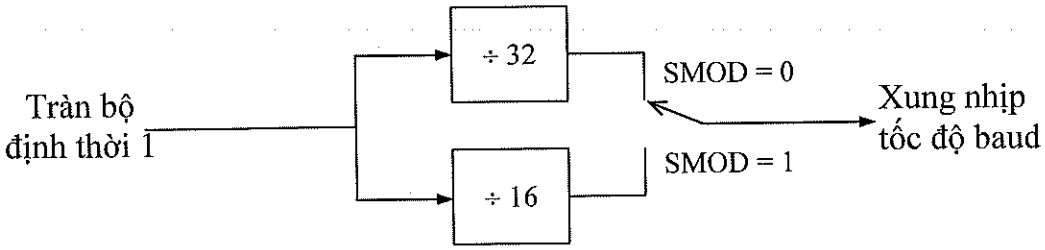
Chế độ 0

Một ứng dụng của chế độ thanh ghi dịch là mở rộng khả năng xuất của 8951. IC thanh ghi dịch nối tiếp ra song song có thể được nối vào các đường TxD và RxD của 8951 để cung cấp thêm 8 đường ra. Có thể nối thêm các thanh ghi dịch với thanh ghi dịch đầu tiên để mở rộng thêm.



3.5.2. Chế độ 1 –UART 8 bit tốc độ thay đổi :

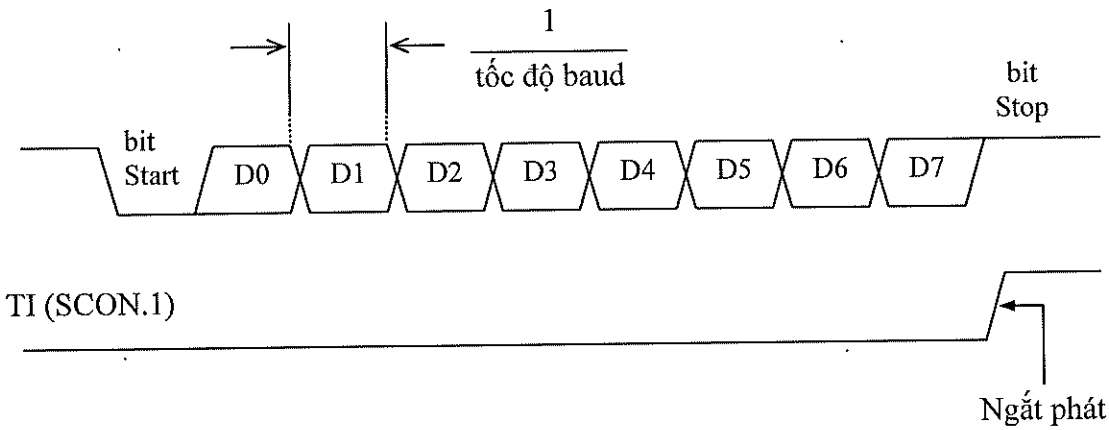
Ở chế độ 1, port nối tiếp của 8951 làm việc như một UART 8 bit (8-bit Universal Asynchronous Receiver/Transmitter) với tốc độ baud thay đổi được (bằng tốc độ báo tràn của Timer 1).



Chế độ 1

Việc thu/phát tiến hành với 10 bit, gồm 1 bit Start (luôn là 0), 8 bit dữ liệu (LSB đầu tiên) và 1 bit Stop (luôn là 1).

Việc phát dữ liệu được khởi động bằng cách ghi dữ liệu cần phát vào sbuf. Dữ liệu được dịch bit để được xuất ra trên đường TxD, bắt đầu bằng bit start, tiếp theo là 8 bit dữ liệu cuối cùng là bit stop. Thời gian của mỗi bit là giá trị nghịch đảo của tốc độ baud. Cờ ngắt phát được set bằng 1 ngay khi bit stop xuất hiện trên đường TxD.



Việc thu dữ liệu được khởi động bằng một chuyển trạng thái từ 1 xuống 0 trên đường RxD (bắt đầu bit start). Bộ đếm 16 tức thời được xóa để đồng bộ số đếm với luồng bit đến. Luồng bit đến được lấy mẫu giữa 16 lần đếm. Giả sử phát hiện được bit Start hợp lệ, bit Start được bỏ qua và 8 bit dữ liệu được đưa vào thanh ghi dịch cổng nối tiếp theo xung nhịp. Khi đã có được tất cả 8 bit, điều sau đây xảy ra:

Bit thứ 9 (bit Stop) được chốt vào RB8 trong SCON.

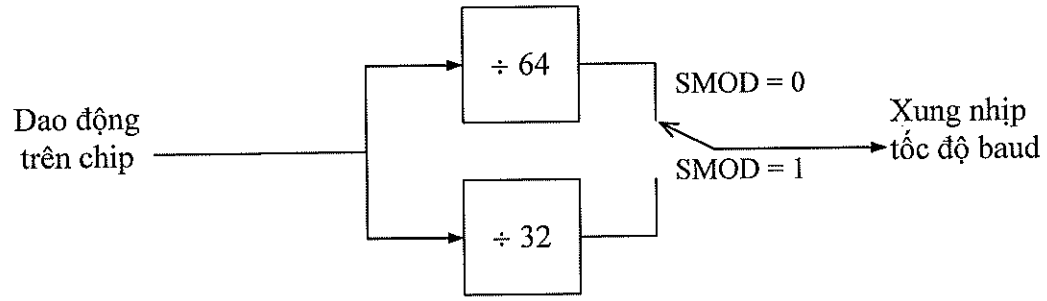
SBUF được nạp với 8 bit dữ liệu.

Cờ ngắt bộ thu (RI) được đặt lên 1 (cờ RI đã xóa về 0 trước khi ký tự được thu về).

3.5.3. Chế độ 2 –UART 9 bit tốc độ cố định :

Ở chế độ 2, port nối tiếp của 8951 làm việc như một UART 9 bit có tốc độ baud cố định (bằng 1/32 hoặc 1/64 tốc độ dao động trên chip). 11 bit sẽ được phát hoặc thu gồm: 1 bit Start, 8 bit dữ liệu, bit dữ liệu thứ 9 có thể lập trình được và 1

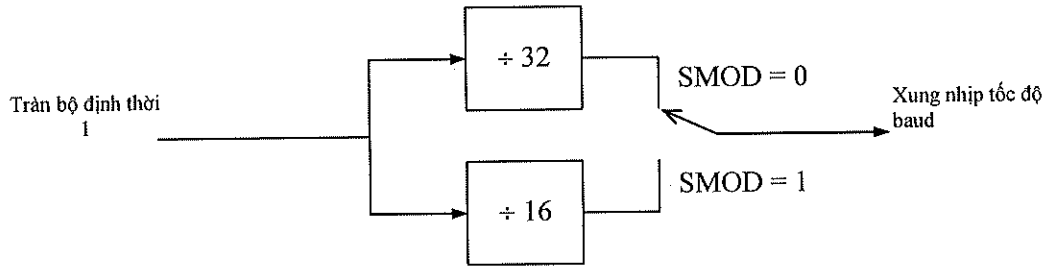
bit Stop. Khi phát, bit thứ 9 là bit bất kỳ được đặt vào bit TB8 trong thanh ghi SCON (có thể là bit chẵn lẻ). Khi thu, bit thứ 9 thu được sẽ ở trong RB8.



Chế độ 2

3.5.4. Chế độ 3 –UART 9 bit tốc độ thay đổi :

Chế độ này giống như chế độ 2, ngoại trừ tốc độ baud có thể lập trình được theo tốc độ tràn của bộ định thời.



Chế độ 3

Thanh ghi điều khiển nguồn cung cấp PCON và bit SMOD:

Tốc độ baud bị ảnh hưởng bởi một bit trong thanh ghi điều khiển nguồn cung cấp (PCON, hay 87H). Bit 7 của PCON là bit SMOD. Đặt bit SMOD lên 1 làm gấp đôi tốc độ baud trong các chế độ 1, 2 và 3.

Vì PCON không được định địa chỉ bit nên để đặt bit SMOD lên 1 có thể thực hiện gián tiếp qua một thanh ghi định địa chỉ bit.

Ví dụ:

```
MOV   A, 87H          ; Lấy giá trị hiện thời của PCON
SETB  ACC.7           ; Đặt bit 7 (SMOD) lên 1
MOV   87H, A          ; Ghi giá trị ngược về PCON.
```

Sử dụng Timer 1 làm xung nhịp tốc độ baud:

Cách thông dụng để tạo tốc độ baud là khởi động bộ định thời 1 ở chế độ 8 bit tự động nạp lại (chế độ 2) và đặt giá trị nạp lại vào TH1 để tốc độ tràn đúng với tốc độ baud. Cũng có thể đạt được các tốc độ baud thấp bằng cách sử dụng bộ định thời ở chế độ 1, tuy nhiên các thanh ghi TH1 và TL1 phải được khởi động lại sau mỗi lần tràn (Việc này được thực hiện trong chương trình phục vụ ngắt). Một chọn lựa khác là cấp xung nhịp cho Timer 1 từ ngoài qua T1 (P3.5).

Công thức tổng quát để xác định tốc độ baud trong các chế độ 1 và 3 (SMOD = 0):

Tốc độ baud = Tốc độ tràn của bộ định thời 1 ÷ 32

Tóm tắt một số chuẩn tốc độ baud thông dụng

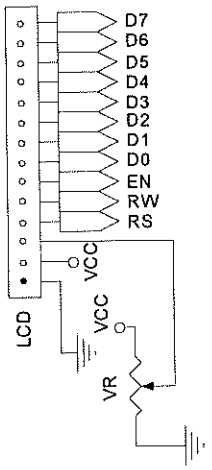
Tốc độ Baud (bps)	Tần số (MHZ)	SMOD (PCON.7)	Giá trị nạp vào TH1	Tốc độ Baud thực (bps)	Sai số (%)
4800	12	0	-7(F9H)	4464	7
9600	12	0	-3(FDH)	10417	8.51
2400	12	1	-26(E6H)	2404	0.16
4800	12	1	-13(F3H)	4808	0.16
9600	12	1	-7(F9H)	8929	7
19200	12	1	-3(FDH)	20833	8.5
4800	11.059	0	-6(FAH)	4800	0
9600	11.059	0	-3(FDH)	9600	0
4800	11.059	1	-12(F4H)	4800	0
9600	11.059	1	-6(FAH)	9600	0
19200	11.059	1	-3(FDH)	19200	0

4. Thiết bị hiển thị LCD :

4.1. Giới thiệu thiết bị :

Màn hình tinh thể lỏng LCD (Liquid-crystal display) lí tưởng cho các ứng dụng công suất thấp. LCD dạng ma trận điểm được sử dụng để hiện thị các mẫu tự chữ và số và các ký hiệu khác. Các loại LCD này được dùng trong điện thoại di động, máy tính số học, máy bán hàng và rất nhiều thiết bị với mục đích cung cấp cho người dùng những thông tin đơn giản. LCD sử dụng trong đề này thuộc loại 2 hàng, mỗi hàng 16 ký tự.

4.2. Sơ đồ chân và chức năng :



Số chân	Ký hiệu	Mức	Chức năng	
1	GND		0V	Nguồn cấp
2	VCC	-	5V	
3	VL	-	-	
4	RS	H/L	H : dữ liệu hiển thị L : dữ liệu điều khiển	
5	R/W	H/L	H : Đọc dữ liệu từ LCD L : Ghi dữ liệu vào LCD	
6	E	H, H→ L	Chân cho phép	
7-14	D0 – D7	H/L	Bus dữ liệu	

LCD có thể giao tiếp dễ dàng với 89C51 như sau : 3 chân từ 4-6 sẽ được nối với 3 chân bất kì của 89C51, bus dữ liệu 7-14 sẽ được nối với một port bất kì của 89C51.

4.3. Các mã lệnh và trình tự điều khiển :

Để giao tiếp với LCD, cần tuân theo những thủ tục nhất định thông qua việc xuất các dữ liệu điều khiển (mã lệnh) cho LCD. Các dữ liệu điều khiển sẽ có các mã riêng biệt dùng cho việc điều khiển các chế độ hiển thị và con trỏ của LCD. Khi đó RS, R/W phải ở mức thấp, đồng thời chân EN phải chuyển từ trạng thái cao

xuống thấp để LCD có thể biết thời điểm nhận dữ liệu. Khi đã điều khiển được các chế độ hiển thị, có thể chọn địa chỉ con trỏ và xuất dữ liệu hiển thị (mã ASCII) ra LCD. Lúc này, R/W = 0 nhưng RS = 1. R/W . R/W = 1 được sử dụng khi ta đọc cờ bận của LCD (chân D7). Chỉ khi cờ bận này bằng 0 thì ta mới có thể xuất dữ liệu hiển thị.

Một trình tự giao tiếp LCD thường dùng được cho trong bảng sau :

1	POWER ON	Cấp nguồn
2	WAIT TILL VCC IS 4.5V MIN	Chờ khi nguồn ít nhất 4.5 V
3	RS=0 ,WRITE 38H	Chọn giao tiếp qua 8bit dữ liệu ,2 hàng, font 5*8 dot
4	RS=0 ,WRITE 0EH	Hiển thị màn hình,con trỏ,con trỏ không chớp
5	RS=0 ,WRITE 06H	Sau mỗi lần viết con trỏ tăng,màn hình không dịch
6	RS=0 ,WRITE 02H	Chuyển con trỏ về đầu hàng 1
7	RS=0 ,WRITE 01H	Xoá màn hình
8	RS=0 ,WRITEADDR	Chọn địa chỉ con trỏ
9	RS=1, WRITE DATA	Xuất ký tự ra màn hình

Địa chỉ con trỏ trên màn hình 2 x 16:

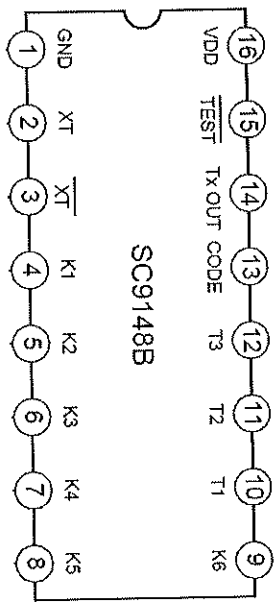
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
80	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF

5. IC ứng dụng trong điều khiển từ xa :

5.1. Giới thiệu IC SC9148 :

SC9148 là IC CMOS được sử dụng cho các bộ điều khiển phát hồng ngoại. SC9148 có thể phát 75 lệnh. Tổng cộng 18 phím ấn có thể được nối vào IC , trong đó có 12 phím đơn và 6 phím liên tục. Nguồn hoạt động từ 2.2V đến 5V. Tần số dao động nội từ 400 đến 600 có thể được tạo ra đơn giản bằng thạch anh và tụ nối ngoài.

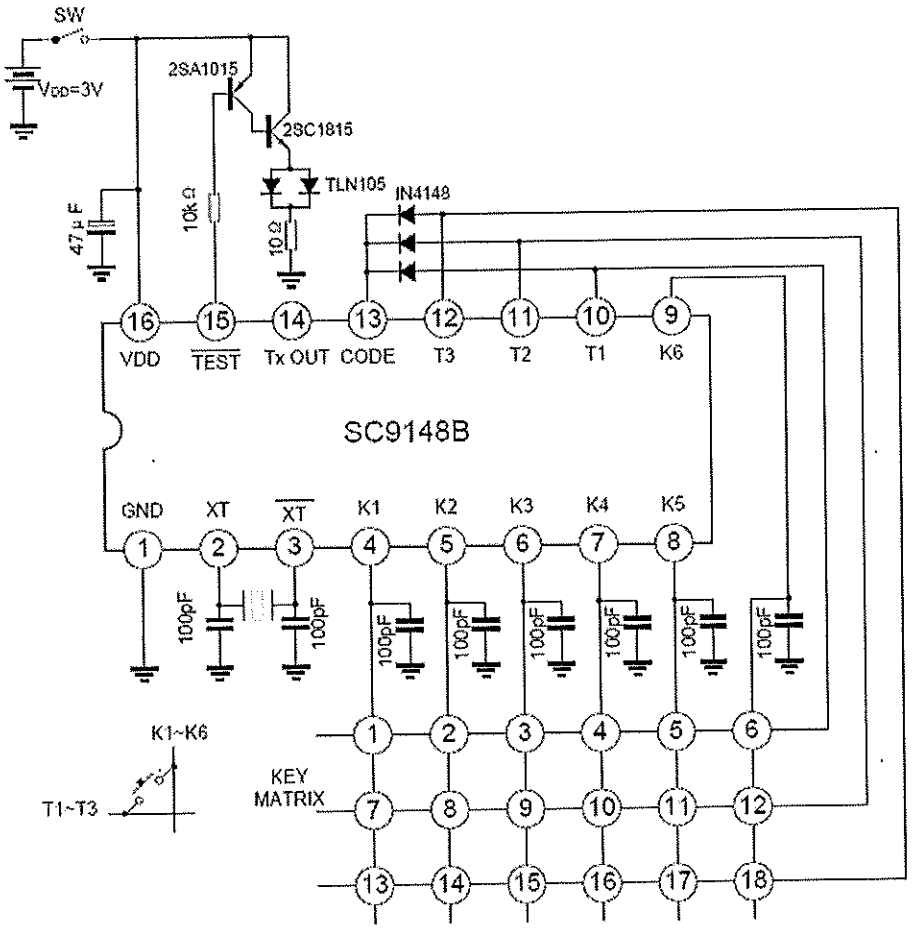
5.1.1. Sơ đồ chân và chức năng :



Số chân	Ký hiệu	Chức năng
1, 16	GND, VDD	Nguồn cấp
2,3	XT, non -XT	Đầu vào ra của bộ dao động
4 - 9	K1-K6	Nhận tín hiệu từ các phím
10-12	T1-T3	Các chân của bộ định thời bên trong
13	CODE	Dùng cho phân biệt các model phát, thu
14	Tx Out	Tín hiệu ra
15	Non -Test	Để hở

5.1.2. Sơ đồ mạch ứng dụng và nguyên lý hoạt động :

Sơ đồ mạch ứng dụng :



Nguyên lý hoạt động :

Như đã giới thiệu, SC 9148 có thể nối với 6 phím liên tục và 12 phím đơn

Đối với phím đơn, khi được ấn, khung dữ liệu chuẩn chỉ được phát một lần. Khung dữ liệu sẽ không phát cho đến khi lần ấn thứ 2. Các phím đơn sẽ được nối vào T2, T3.

Đối với phím liên tục, khi được ấn , khung dữ liệu sẽ được phát lập lại 1 các liên tục cho đến khi phím được buông. Các phím liên tục sẽ được nối vào T1.

Khung dữ liệu phát đi có cấu trúc như sau :

C1	C2	C3	H	S1	S2	D1	D2	D3	D4	D5	D6

✓ C1, C2, C3 : là các bit CODE phụ thuộc vào cách đấu nối T1, T2, T3 vào chân CODE.

- Nếu sử dụng SC9149A để thu, C2 và C3 sẽ được sử dụng. Nếu sử dụng SC 9150A để thu, C1 và C2 sẽ được sử dụng.
- Bit C_i sẽ là 1 khi chân T_i được nối vào chân CODE qua diode, bit C_i sẽ là 0 khi T_i bỏ hở. Lưu ý, trong 4 trạng thái , trạng thái (0,0) không được sử dụng cho các cặp bit C1,C2 và C2,C3. Cách nối T_i vào chân CODE và cách nối các chân $CODE_i$ ở IC thu phải tạo ra các cặp bit giống nhau.
- Do IC thu chỉ có các chân CODE 1, CODE 2 (SC9150A), hoặc CODE 2, CODE 3 (SC9149A), các CODE 3 và CODE 1 không có nhưng được mặc định là 1. Vì vậy, ở đầu phát , T3 và T1 cũng phải được nối vào chân CODE.

✓ H (Hold) : Khi các phím liên tục được ấn, H sẽ là 1

✓ S1,S2 (Single1, Single2) : Khi các phím đơn được ấn, S1 hoặc S2 sẽ là 1.

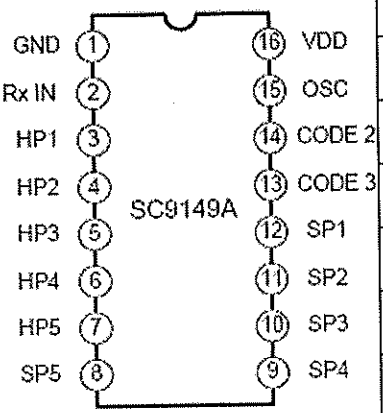
(Thứ tự ưu tiên là T1, T2, T3 , vì vậy luôn chỉ có H hoặc S1 hoặc S2 là 1.)

✓ D1-D6 : khi các phím nối với K_i được ấn, D_i sẽ là 1. Việc ấn các tổ hợp phím liên tục sẽ tạo ra các dữ liệu D1-D6 khác nhau. Như vậy, sẽ có 2^6-1 trạng thái (bỏ trạng thái "000000") tương ứng với 63 lệnh tạo ra. Các phím đơn không thể tạo tổ hợp nên chỉ có 12 lệnh tạo ra. Vậy , SC9148 có thể phát tối đa là 75 lệnh.

5.2. Giới thiệu IC SC9149 :

SC9149A/ SC9150A là IC CMOS được thiết kế cho các bộ điều khiển thu hồng ngoại, khi kết hợp với SC9148 có thể tạo ra 1 hệ thống điều khiển từ xa. SC9150A có thể thu 75 lệnh từ SC9148 nhưng SC9149A chỉ thu được 36 lệnh.

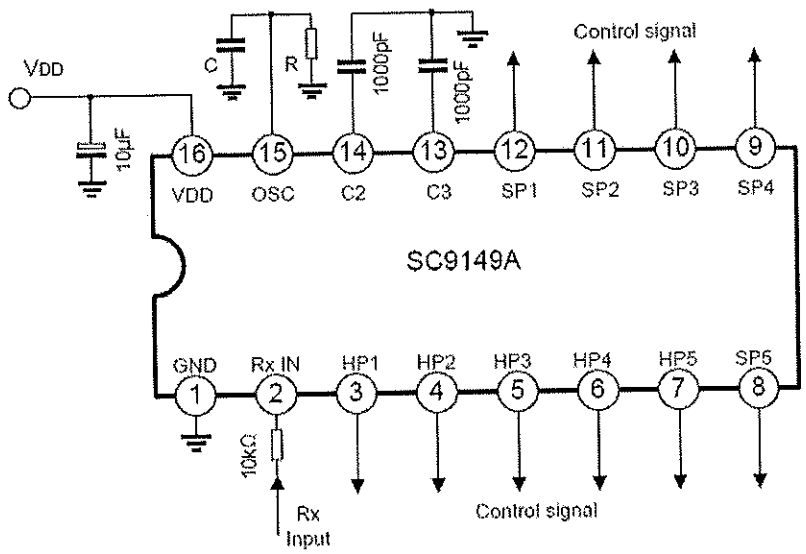
5.2.1. Sơ đồ chân và chức năng :



Số chân	Ký hiệu	Chức năng
1, 16	GND, VDD	Nguồn cấp
2	Rx IN	Thu tín hiệu hồng ngoại
3-7	HP1-HP5	Lên 1 khi các phím liên tục được ấn
8-12	SP5- SP1	Lên 1 khi các phím đơn được ấn
13-14	CODE2-CODE3	Dùng cho phân biệt các model thu
15	OSC	Tạo dao động

5.2.2. Sơ đồ mạch ứng dụng và nguyên lý hoạt động :

Sơ đồ mạch ứng dụng :



Nguyên lý hoạt động :

Bit C_i khung dữ liệu sẽ là 1 khi $CODE_i$ được nối xuống GND qua tụ , C_i sẽ là 0 khi $CODE_i$ được nối GND. Như vậy, sơ đồ trên sẽ giải mã các tín hiệu phát có C2 = C3 = 1.

Nếu $_{giao}$ tiếp với 89C51, các chân SP1-SP5, HP1- HP5 sẽ được nối vào các port. 89C51 sẽ đọc các chân này để phát lệnh điều khiển đến các thiết bị tương ứng.

HP1-HP5 sẽ tạo ra các tổ hợp của 5 bit khác nhau, cụ thể là $2^5 - 1 = 31$ tổ hợp và SP1- SP5 chỉ tạo được 5 tổ hợp. Vậy số lệnh có thể thu được là 36 lệnh.

6. Giao tiếp máy tính và 89C51 :

6.1. Giới thiệu chuẩn giao tiếp RS-232 :

RS-232 là một trong các chuẩn giao tiếp phổ biến nhất. Hiện nay, đã có chuẩn giao tiếp nhanh hơn, phức tạp hơn nhưng RS-232 vẫn tiếp tục phổ biến do phần cứng và phần mềm đơn giản và không mắc tiền, và vì đa số các thiết bị hiện tại đều đã có giao diện này tích hợp vào.

RS-232 được thiết kế cho việc truyền thông giữa 2 thiết bị với khoảng cách khoảng 15m đến 150m phụ thuộc vào tốc độ truyền và loại cáp.

Mức logic cao : -12 V, mức logic thấp : +12 V

Hai thuật ngữ thường dùng trong RS-232 là DTE (Data Terminal Equipment) và DCE (Data Circuit-terminating Equipment). TD được định nghĩa là tín hiệu ra của DTE và tín hiệu vào của DCE còn RD thì ngược lại. Các cổng nối tiếp trên PC được cấu hình là DTE và các cổng trên modem là DCE. Hầu hết các thiết bị ngoại vi còn lại là DCE.

RS-232 nếu không sử dụng các cơ chế bắt tay thông qua các chân điều khiển thì chỉ cần 3 dây cho việc phát thu dữ liệu :

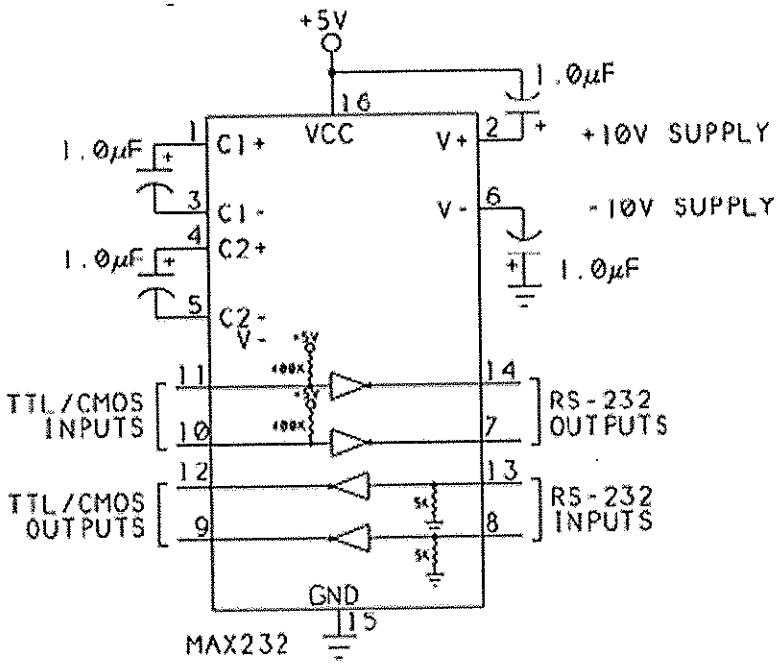
TD : mang dữ liệu từ DTE tới DCE , còn được gọi là TX và TXD

RD : mang dữ liệu từ DCE đến DTE , còn được gọi là RX và RXD

SG (signal ground) : điểm so của của tín hiệu, còn được gọi là GND hoặc SGND

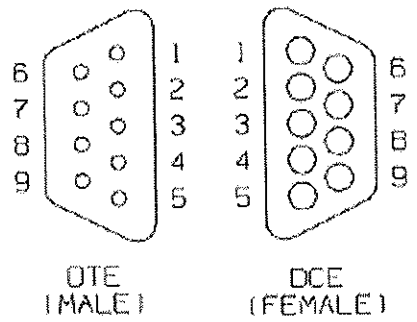
6.2. Giới thiệu IC MAX 232 :

Maxim Semiconductor là công ty đầu tiên đưa ra các chip giao diện RS-232 chỉ cần nguồn +5V. MAX232 gồm 2 bộ lái chuyển đổi các đầu vào TTL thành đầu ra RS-232. Sơ đồ hoạt động của MAX232 như sau :



6.3. Đầu nối và sơ đồ kết nối :

Các đầu nối gồm loại 9 chân (DB9) và 25 chân (DB25). Các DTE sử dụng đầu nối đực và DCE sử dụng đầu nối cái. Thứ tự chân của đầu nối DB9 (thường sử dụng hơn) được cho trên hình sau :



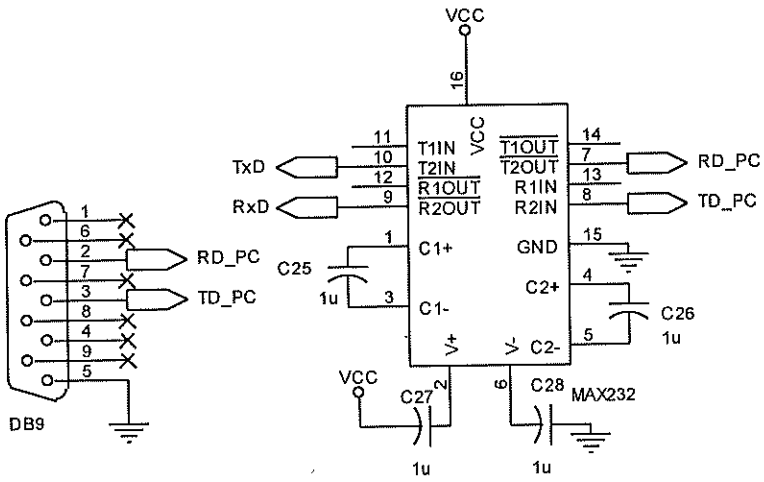
Nếu không sử dụng các cơ chế bắt tay, ta chỉ quan tâm đến 3 chân :

2 : RD

3 : TD

5 : GND

Sơ đồ kết nối cho 1 cặp phát thu dữ liệu ở DCE được cho trên hình sau:

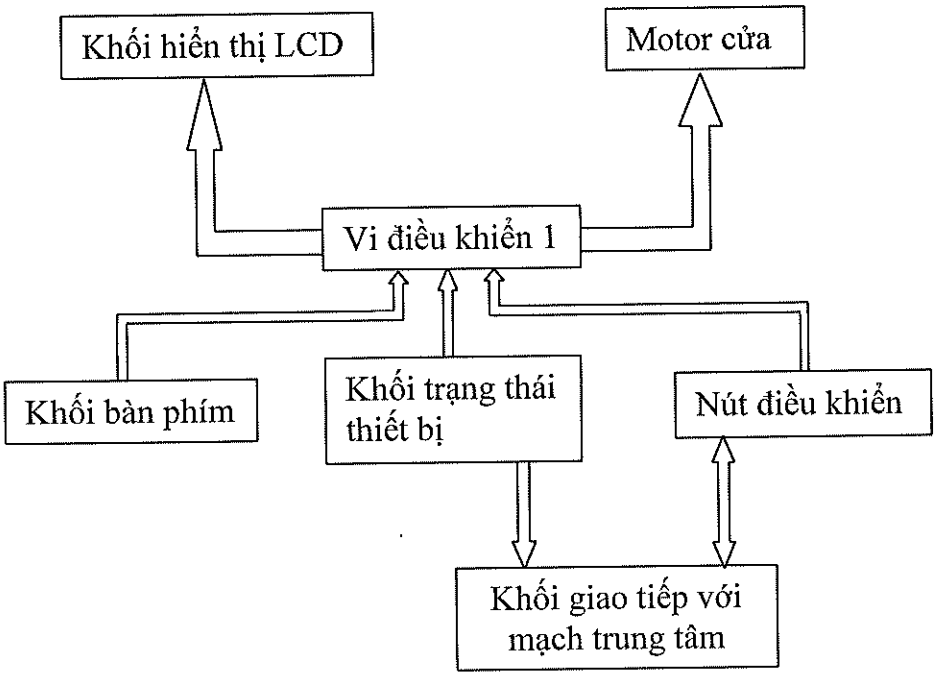


TxD, RxD : là chân phát thu dữ liệu của DCE ví dụ như 89C51

PHẦN 3. TÍNH TOÁN – THIẾT KẾ MẠCH

1. Mạch điều khiển cửa :

1.1. Sơ đồ khối và chức năng các khối :



- ✓ Khối bàn phím : nhập password phù hợp để mở cửa
- ✓ Khối hiển thị LCD : hiển thị các dòng text theo lệnh từ vi điều khiển
- ✓ Khối vi điều khiển 1 : nhận các tín hiệu từ khối bàn phím, khối trạng thái thiết bị và nút điều khiển để :
    - o **Điều khiển LCD :**
        - Hiển thị 2 lựa chọn cho người dùng : nhập password hoặc ấn chuông.
        - Nếu chọn nhập password , các số khi nhập sẽ hiển thị. Nếu nhập sai thì hiển thị lại 2 lựa chọn ban đầu còn nếu nhập đúng sẽ hiện thị dòng chữ ' WELCOME HOME'. Sau đó hiển thị dòng chữ " CLOSE THE DOOR, PLEASE". Dòng chữ sẽ hiển thị cho đến khi cửa được đóng. Khi cửa đã đóng, LCD hiển thị 2 lựa chọn ban đầu và quá trình được lặp lại.

- Nếu ấn phím chuông, màn hình hiển thị "BELL IS RINGING…." trong một khoảng thời gian ngắn rồi hiển thị lại 2 lựa chọn ban đầu. Nếu trong 20s đó cửa được mở, hiển thị "CLOSE THE DOOR, PLEASE". Dòng chữ sẽ hiển thị cho đến khi cửa được đóng. Khi cửa đã đóng, LCD hiển thị 2 lựa chọn ban đầu và quá trình được lặp lại
- Nếu ấn phím đóng/mở cửa , hiển thị dòng chữ " CLOSE THE DOOR, PLEASE"(nếu trước đó cửa đóng). Dòng chữ sẽ hiển thị cho đến khi cửa được đóng. Khi cửa đã đóng, LCD hiển thị 2 lựa chọn ban đầu và quá trình được lặp lại.

o Điều khiển motor : để đóng hoặc mở cửa (điều khiển chiều motor)

✓ Khối trạng thái thiết bị :

o Gửi tín hiệu từ công tắc hành trình gắn trên cửa cho vi điều khiển 1 để điều khiển motor.

o Gửi tín hiệu khi cửa đã được đóng sát để vi điều khiển 1 và 2 biết trạng thái cửa.

✓ Khối nút điều khiển : gồm nút chuông, nút đóng/mở cửa, nút chọn nhập password.

o Nút chuông : khi được ấn lúc LCD hiển thị lựa chọn sẽ phát tín hiệu đến vi điều khiển 2 trong nhà , rồi truyền đến PC để kích hoạt âm thanh chuông trên loa PC.

o Nút nhập password : xác nhận lựa chọn nhập password khi LCD hiển thị lựa chọn.

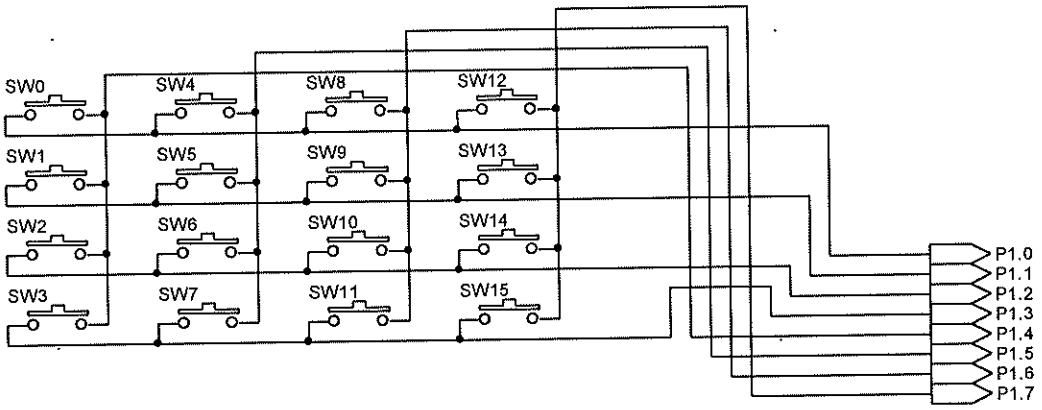
o Nút đóng mở cửa : để người phía trong nhà có thể thao tác đóng/ mở, nếu cửa đang đóng, ấn phím cửa sẽ mở và ngược lại. Nút đóng mở cửa cũng được điều khiển bởi vi điều khiển 2 đặt trong nhà.

- ✓ Khối giao tiếp với mạch điều khiển trung tâm : truyền tín hiệu chuông, trạng thái cửa cho vi điều khiển 2 và nhận tín hiệu đóng/mở cửa thông qua cáp nối với mạch điều khiển trung tâm. .

1.2. Tính toán thiết kế :

1.2.1. Khối bàn phím :

Sử dụng bàn phím ma trận 16 phím. Password được cài đặt có thể đến 16 kí tự chữ và số. Sơ đồ như sau :

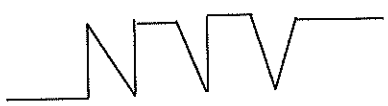


P1.0 đến P1.3 được xem như các chân điều khiển hàng, P1.4 đến P1.7 được xem như các chân điều khiển cột.

Để biết được trạng thái các phím đã buông ra chưa, ta sẽ cho các điều khiển cột ở mức 0 và điều khiển hàng ở mức 1. Tín hiệu sẽ được đọc vào điều khiển hàng, nếu có bất kỳ phím nào ấn dữ liệu đọc vào sẽ khác trạng thái 1111.

Để biết trạng thái phím nào được ấn, ta lần lượt cho từng cột ở mức 0, ứng với mỗi lúc có một cột bằng 0, ta lại cho từng hàng ở mức 1 và đọc dữ liệu hàng vào . Đánh số thứ tự cho phím theo thứ tự kiểm tra của ta (có 4x4=16 lần kiểm tra). Lần kiểm tra nào có bit đọc vào là 0 tương ứng với phím có số thứ tự của lần kiểm tra đó được ấn.

Các phím được ấn thường xảy ra hiện tượng nảy trong khoảng 10 – 20 ms đầu gây ra hiện tượng đọc sai dữ liệu:



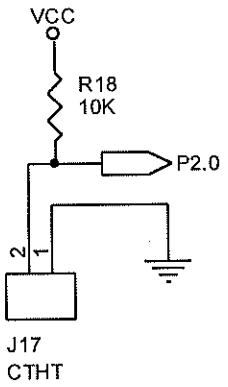
Trong thiết kế không sử dụng mạch chống nảy bằng phần cứng, phần chống nảy sẽ được thực hiện bằng phần mềm. Khi lập trình đọc trạng thái các phím, ta chỉ cần chờ khoảng 10 – 20 ms rồi mới đọc dữ liệu là khắc phục được.

1.2.2. Khối hiển thị LCD :

Trình tự giao tiếp và sơ đồ kết nối đã được nói ở phần 2 mục 4

1.2.3. Khối trạng thái thiết bị :

Công tắc hành trình (CTHT) thường được sử dụng để điều khiển chiều hoặc trạnng thái của motor. Một CTHT chỉ đơn giản là 2 lá nhôm, bình thường thì không chạm vào nhau. Khi cửa tịnh tiến đến một vị trí nào đó được qui định, (chẳng hạn của đã đóng sát) thì 2 lá nhôm sẽ chạm vào nhau và ta phải biết được thời điểm này. Vậy để đọc được trạng thái một CTHT, ta chỉ cần một mạch đơn giản sau :

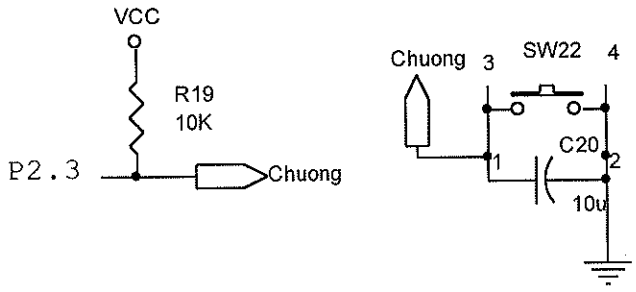


Bình thường, P2.0 ở mức 1, khi CTHT on , P2.0 sẽ chuyển về mức 0.

Để phát hiện trạng thái cửa đã được đóng hay chưa, ta cũng có một mạch tương tự.

1.2.4. Khối nút điều khiển :

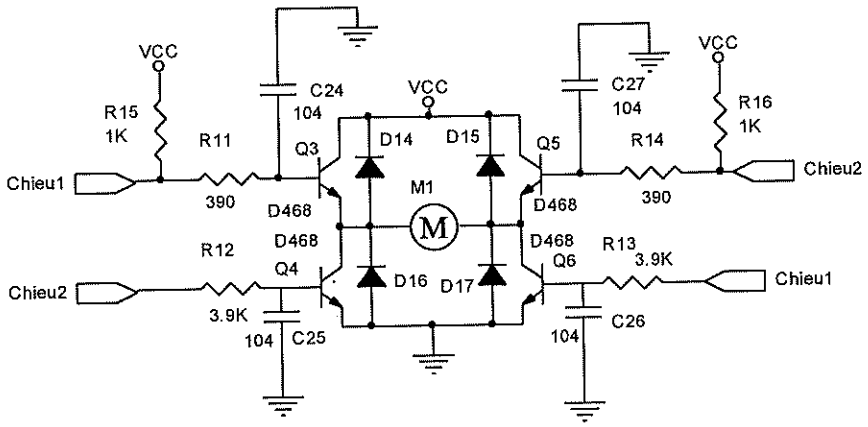
Các nút điều khiển đều có chung một dạng mạch như sau :



Các tụ dùng chống nảy cho công tắc. Điện trở kéo lên làm giảm giá trị điện trở kéo lên nội trong chip 89C51 và duy trì trạng thái logic 1 trên một đường truyền dài. Khi chuông được ấn, P2.3 và đường truyền sẽ xuống mức thấp và vi điều khiển 2 sẽ phát hiện được thời điểm này để truyền tín hiệu báo chuông cho PC.

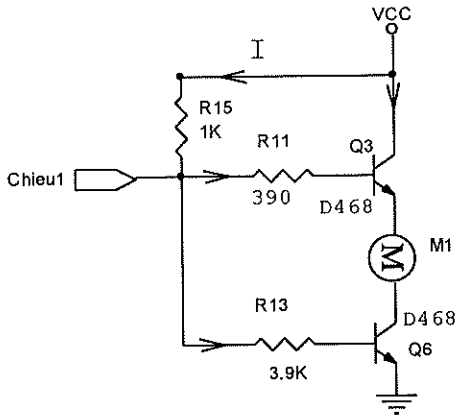
1.2.5. Khối motor :

Motor và cửa mô hình sử dụng được lấy từ cơ cấu đóng mở của một ổ CDROM cũ. Motor một chiều 5V, dòng tải khoảng 140mA. Chiều của motor được điều khiển từ một mạch cầu gồm 4 transitor NPN D468.



Khi chiều 1 ở mức 1, cặp transitor Q_3 và Q_6 sẽ dẫn, khi chiều 2 ở mức 1, cặp transitor Q_4 và Q_5 sẽ dẫn. Theo sơ dồ mạch, mỗi một cặp transitor khi dẫn sẽ làm thay đổi cực tính của nguồn cấp cho motor, vì thế ta có thể thay đổi chiều của motor. Các diode để giảm bớt các xung nhọn tạo ra bởi các cuộn dây motor và bảo vệ các linh kiện trong mạch. Để cửa mô hình được đóng mở với tốc độ vừa phải, ta thiết kế mạch với điện áp đặt vào motor chỉ 3V và dòng qua nó là I_{C3_6}= 100mA.

Vẽ lại mạch như sau (xét cặp Q_3,Q_6):



Gọi :

I_{B3}, I_{B6}, I_{C3_6}, V_{CE3}, V_{CE6}, V_{BE3}, V_{BE6} lần lượt là dòng I_B , I_C và áp V_{CE}, V_{BE} trên các transitor Q_3 , Q_6 .

I là dòng qua R_{15}

Ta có 2 phương trình Kirchhoff :

$$V_{CC} = I R_{15} + I_{B6} R_{13} + V_{BE6}$$

$$\Leftrightarrow R_{13} = \frac{V_{CC} - V_{BE6} - IR_{15}}{I_{B6}} \geq 0 \quad (1)$$

$$\Leftrightarrow I \leq \frac{V_{CC} - V_{BE6}}{R_{15}} \quad (3)$$

$$V_{CC} = I R_{15} + I_{B3} R_{11} + V_{BE3} + V_{M1} + V_{CE6}$$

$$\Leftrightarrow I R_{15} + I_{B3} R_{11} + V_{BE3} = V_{CE3}$$

$$\Leftrightarrow R_{11} = \frac{V_{CE3} - V_{BE3} - IR_{15}}{I_{B3}} \geq 0 \quad (2)$$

$$\Leftrightarrow I \leq \frac{V_{CE3} - V_{BE3}}{R_{15}} \quad (4)$$

Giả sử $V_{BE3} = V_{BE6}$, để $R_{11}, R_{13} \geq 0$ ta chỉ cần lấy điều kiện (4)

Theo datasheet của AT89C51, dòng I_{OLmax} = 10mA, do đó điện trở kéo lên R_{15} > V_{CC} / I_{OLmax} = 5/10 = 0.5 KΩ . (*)

Nếu chọn transitor C1815 thiết kế, điện trở R_{15} tính được sẽ rất nhỏ khoảng 100 ohm, không thỏa điều kiện trên , hơn nửa C1815 có I_{CMAX} chỉ là 150mA không đảm bảo cho hoạt động an toàn. Vì vậy ta chọn transitor D468 có các thông số như sau :

I_{CMAX} = 1 A

β = 85 - 240

$V_{CESAT\,MAX}$ = 0,5 V

$V_{BESATMAX}$ = 1V

Theo (*), chọn R_{15} = 1K, V_{CE3} = 2V, V_{BE3} = 0,7V, từ (4) có điều kiện

$$I \leq \frac{2-0,7}{1} = 1,3 \text{ mA}$$

$\Leftrightarrow I_{B3} + I_{B6} < 1,3$ mA

Hệ số khuyếch đại của Q3 được đo thực tế là khoảng 280 ứng với VCE3 = 2V và IC3 = IC3_6 = 100 mA.

=> IB3 = 100/ 280 = 0,35 mA

Chọn IB2 = 0.8 mA > IC / β để transitor Q6 bão hòa

=> I = 0,8 + 0,35 = 1,15 mA. Thế vào (1) và (2), tính được

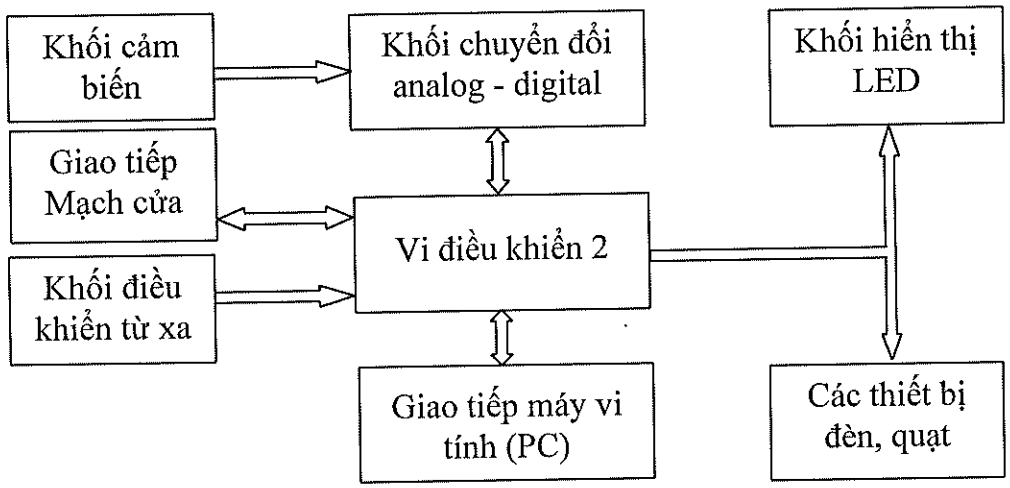
R_{13} = 3,93 KΩ

R_{11} = 428 Ω .

Chọn R_{13} = 3.9 KΩ ; R_{11} = 390 Ω

2. Mạch điều khiển trung tâm :

2.1. Sơ đồ khối và chức năng các khối :



- ✓ Khối cảm biến : gồm 4 cảm biến -2 cảm biến nhiệt và 2 cảm biến ánh sáng được gắn ở 2 phòng. Dựa trên sự thay đổi nhiệt độ và ánh sáng , các cảm biến sẽ truyền dữ liệu analog cho khối chuyển đổi.
- ✓ Khối chuyển đổi analog-digital : giao tiếp với vi điều khiển và xuất dữ liệu từ cảm biến đã chuyển đổi ra dạng số cho vi điều khiển.
- ✓ Giao tiếp mạch cửa : sử dụng cáp tín hiệu cho giao tiếp giữa vi điều khiển 2 và mạch cửa.
- ✓ Vi điều khiển 2:
    - o sau khi nhận dữ liệu từ khối chuyển đổi analog-digital, xuất nhiệt độ phòng 1 ra khối hiển thị LED.
    - o Sau khi nhận tín hiệu điều khiển thiết bị từ khối điều khiển từ xa, sẽ phát tín hiệu điều khiển các thiết bị tương ứng.
    - o Nhận tín hiệu trạng thái cửa, chuông từ khối giao tiếp mạch cửa , truyền tín hiệu đóng/mở cửa từ PC đến mạch cửa qua khối giao tiếp mạch cửa.
    - o Truyền các dữ liệu về nhiệt độ và trạng thái các thiết bị đèn, quạt kể cả các tín hiệu nhận được từ mạch cửa cho PC.

- Nhận và thi hành các lệnh điều khiển cửa và các thiết bị đèn, quạt từ PC.

✓ Máy vi tính :

- Nhận các dữ liệu về trạng thái thiết bị, cửa từ vi điều khiển 2 và hiển thị trên giao diện phần mềm.
- Từ giao diện phần mềm, có thể điều khiển các thiết bị .

2.2. Tính toán thiết kế :

2.2.1. Khối cảm biến nhiệt độ :

Với các dụng điện tử sử dụng vi xử lí/ vi điều khiển, các IC cảm biến nhiệt độ chuyên dụng với kích thước nhỏ, độ chính xác khá cao là một trong những lựa chọn tốt nhất. Trong đề tài sẽ sử dụng IC LM35 do không cần mạch cân chỉnh, sai số thấp, giá thành không mắc.

Như đã giới thiệu, LM 35 có thể đạt được sai số ± 0.4 oC tương đương với mức điện áp 5 mV(10mV / °C). Với điện áp Vref = 1.28 V, độ phân giải điện áp của ADC 0809 là 1.28/256 = 5 mV. Tuy nhiên trong quá trình thi công, ADC không hoạt động đúng ở dãy điện áp 700mV đến 1V với Vref 1.28V. Giá trị Vref để ADC hoạt động đúng là khoảng từ 2 V trở lên. Vì vậy, để phần mềm gọn nhẹ, ta sẽ chọn Vref cho ADC là 2.56V tương ứng với độ phân giải 10mV. Để tính sai số cho hệ thống đo, gọi:

x (mV) là giá trị điện áp thật của của đầu ra LM35

x_T (mV) là điện áp mà LM35 cần phải đưa ra từ nhiệt độ thật trong phòng

y (mV) là điện áp đọc được từ ADC 0809 với đầu vào kx.(k : hệ số khuếch đại)

Giả sử LM35 mắc sai số điện áp $\pm$ α mV. Theo datasheet, ADC 0809 mắc sai số tối đa $E_{TUE} = \pm 1$ LSB $\Leftrightarrow$ 2,56V /256 = 10mV. Bỏ qua sai số của mạch khuếch đại, ta có 2 bất đẳng thức sau :

$$x - \alpha < x_T < x + \alpha \quad (1)$$

$$y - E_{TUE} < kx < y + E_{TUE} \quad (2)$$

$$(2) \Leftrightarrow \frac{y}{k} - \frac{E_{TUE}}{k} < x < \frac{y}{k} + \frac{E_{TUE}}{k} \quad (3)$$

Lấy (1) + (3) ta được :

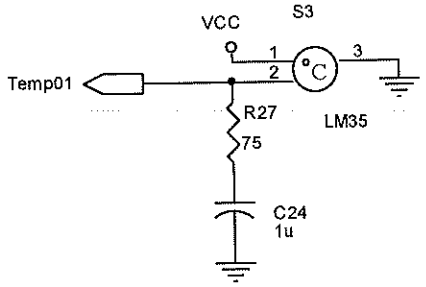
$$\frac{y}{k} - \alpha - \frac{E_{TUE}}{k} < x_T < \frac{y}{k} + \alpha + \frac{E_{TUE}}{k}$$

Từ đẳng thức trên, điện áp sẽ được vi điều khiển qui đổi ra nhiệt độ phòng y/k sẽ mắc phải sai số E_S (mV) = ± ($\alpha + E_{TUE} / k$) . Như đã đề cập ở phần 2, LM35 sai số ±0,4 °C (25 °C) và ± 0,8 ° C (T_{min} , T_{max}), suy ra $\alpha = 4$ hoặc $\alpha = 8$. Ta tính được E_S (°C) như sau :

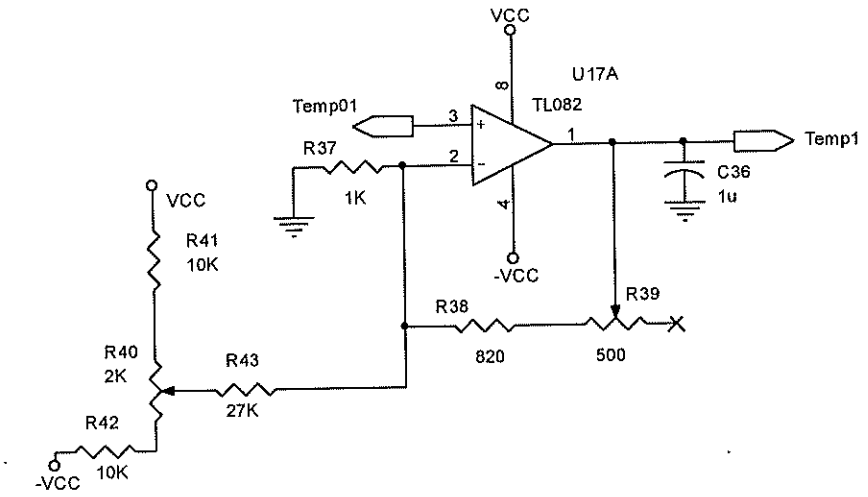
k \ α	4	8
1	±1,4	±1,8
2	±0,9	± 1,3
4	± 0,65	±1,05

Với k = 4, Op amp cần được tăng nguồn lớn hơn 5 V mới đạt được độ khuyếch đại mong muốn trong dải điện áp đo. Trong thiết kế, ta chọn k = 2 với sai số chấp nhận được. Như vậy, tín hiệu ra của cảm biến sẽ được nhân hai trước khi đưa vào ADC.

LM35 có một giới hạn khi lái các tải dung kháng lớn (tạo ra từ các dây nối), hơn nữa, cũng như các mạch điện tuyến tính khác khi được nối dây trong một môi trường với nhiều nhiễu điện từ mạnh từ motor, role, máy phát thanh … hoạt động của LM35 sẽ bị ảnh hưởng nghiêm trọng. Để khắc phục điều này, đầu ra sẽ nối với một điện trở 75 Ohm nối tiếp với tụ 0.2 µF hoặc 1µF xuống GND sẽ cho kết quả tốt nhất :



Mạch nhân hai tín hiệu có thể dùng mạch khuyếch đại không đảo dùng Op Amp TL082 như hình sau :



Hệ số khuyếch đại được cho bởi biểu thức : $A_V = 1 + (R_{38} + R_{39})/R_{37}$

Chọn R_{37}= 1 K, biến trở 500 Ohm dễ dàng cho việc tinh chỉnh, để $A_V = 2$ ⇨ chọn R38 = 820 Ohm cho điểm chỉnh biến trở ở khoảng giữa.

Do có thể làm việc với mức điện áp cỡ chục mV , một mạch chỉnh offset cần thêm vào cho Op Amp. Hai điện trở 10K được gắn hai đầu R_{40} thuận lợi cho việc tinh chỉnh, $R4_3$ có giá trị lớn nhằm tránh phân dòng qua nó. Các giá trị R_{41} , R_{42} , R_{43} có thể chọn như trên hình..

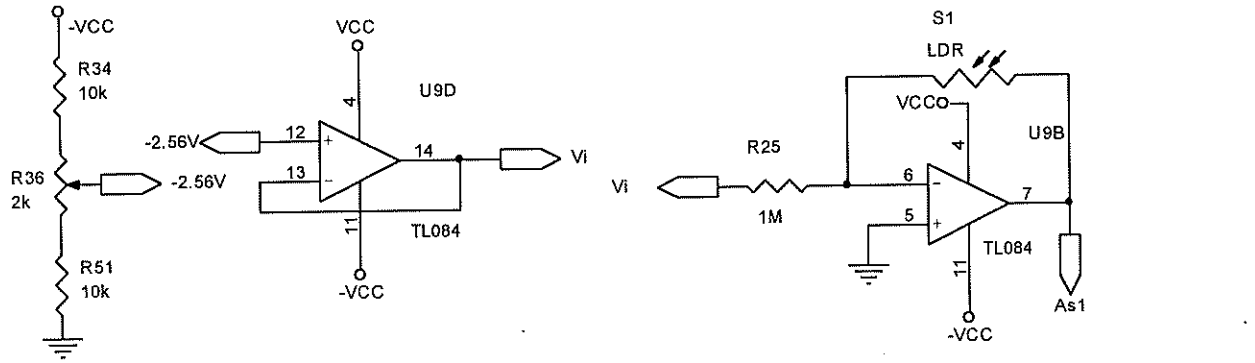
Tụ ở ngõ ra chỉ để lọc điện áp.

2.2.2. Khối cảm biến ánh sáng :

Sử dụng cảm biến ánh sáng chỉ dừng lại ở việc bật tắt các đèn tuỳ theo ánh sáng trong phòng, không thực hiện các mạch chỉnh độ sáng (dimmer) phức tạp nên có thể chọn lựa các cảm biến quang trở nhỏ, giá thành khá rẻ nhưng hoạt động cũng rất chính xác và ổn định.

Để có thể đo được mức ánh sáng trong phòng (hoặc ngoài trời), người thiết kế cần tiến hành đo đạc thực tế điện trở của cảm biến. Ở một cự li xác định trước, độ sáng trong phòng được cho là không đủ khi giá trị điện trở lớn hơn 50 KΩ. Ứng với cự li đó, khi 1 đèn huỳnh quang 1m2 được bật lên , giá trị điện trở giảm xuống còn 30 KΩ. Khi cần bật sáng một đèn trước sân nhà vào chặp tối, điện trở cảm biến khoảng 1 MΩ. Để có thể xác định được độ sáng trong nhà cũng như ngoài trời đúng vào lúc cần bật đèn, ta chỉ cần đo đạc được các giá trị điện trở trên.

Để đo điện trở, ta có thể sử dụng Op Amp làm nguồn dòng và cần một nguồn điện áp chuẩn khác. Mạch đo như sau :



V_i là nguồn điện áp chuẩn, có thể tạo ra bằng mạch lặp điện áp. Các R_{34} , R_{36}, R_{51} được tính tương tự như phần 2.2.3.1 . Dòng vào Op Amp gần bằng 0 nên dòng qua quang trở S1 sẽ không đổi. Nếu gọi R_Q là điện trở của quang trở S1, V_{out} là áp tại chân số 7 của TL084, ta sẽ thu được phương trình sau :

$R_Q = -V_{out} / V_i * R_{25}$ hay $R_{25} = - R_Q * V_i / V_{out}$

Để đo được điện trở 1 MΩ, V_{out} phải bé thua 2,56V (là Vref khi đo nhiệt độ), ta có thể chọn tỉ số $V_i / V_{out} = -1 \Leftrightarrow$ Vi =2,56V để $R_{25} = R_Q =$ 1MΩ. Vậy ứng với V_{out} = 5 mV(giá trị ½ LSB, tại giá trị này ADC sẽ xuất mã 01_{HEX}), thế vào sẽ tính R_Q = 2 KΩ. Vậy khoảng điện trở có thể đo của mạch là từ 2 KΩ đến 1 MΩ, thỏa mãn các yêu cầu đã đặt ra.

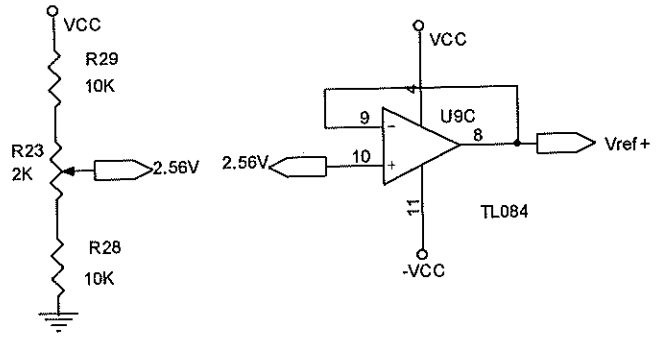
2.2.3. Khối chuyển đổi analog-digital :

Để có thể đọc dữ liệu từ 4 cảm biến, ADC0809 sẽ là sự lựa chọn tốt do có tới 8 ngõ vào.

2.2.3.1. Mạch tạo điện áp chuẩn :

Có thể sử dụng mạch lặp điện áp dùng Op Amp TL084 với trở kháng ra thấp, có thể tạo điện áp chuẩn ổn định. Sơ đồ mạch được cho ở trang kế :

Các điện trở lớn ở 2 đầu mạch chiết áp được dùng để thuận tiện cho việc tinh chỉnh. Chọn biến trở 2K. Để tính R_{28} , R_{29} , tính điện áp ra ứng với điểm chỉnh biết trở ở 2 đầu mút và chọn V_{out_min} = 2,3 V ; V_{out_max} = 2,8 V để có thể chứa giá trị 2,56.(khoảng chọn không quá bé bởi phải tính đến khoảng biến thiên của điện áp nguồn và sai số điện trở):



$$\begin{cases} V_{out_min} = \dfrac{Vcc.R_{28}}{R_{28}+R_{29}+2} = 2,3 \\ V_{out_max} = \dfrac{Vcc.(R_{28}+2)}{R_{28}+R_{29}+2} = 2,8 \end{cases}$$

Với Vcc = 5V, ta sẽ có 2 phương trình :

$$\begin{cases} \dfrac{R_{28}+2}{R_{28}} = \dfrac{2,8}{2,3} = 1,21 \\ 0,78R_{28} + 1,57 = R_{29} \end{cases} \Leftrightarrow \begin{cases} R_{28} = 9,52 \\ R_{29} = 9 \end{cases}$$

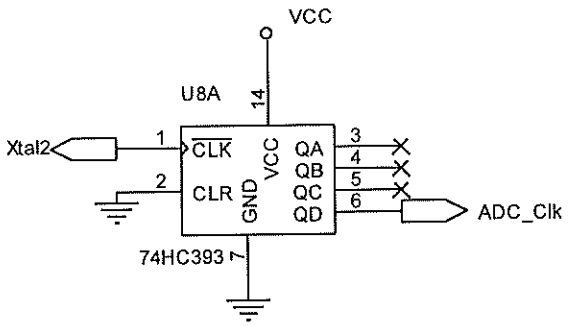
Các giá trị điện trở chuẩn có thể chọn là 8K2 , 10K. Có thể chọn $R_{28} = R_{29} =$ 10K để điện áp ra luôn chứa giá trị 2,56V. Bảng kiểm tra lại như sau :

R_{28}	R_{29}	V_{out_max}	V_{out_min}	Vcc
10	10	2.73	2.27	5
10	10	2.67	2.23	4.9
10	10	2.62	2.18	4.8
10	10	2.78	2.32	5.1
10	10	2.84	2.36	5.2

(V_{out_max} , V_{out_min} cũng thỏa khi xét đến sai số ± 5% của điện trở.)

2.2.3.2. Mạch tạo xung clock :

Mạch tạo xung clock cho ADC có thể dùng các mạch tạo dao động RC kết hợp các cổng logic hoặc cổng đảo trigger Schmitt. Tuy nhiên để ổn định hơn có thể chia tần số 12Mhz từ dao động thạch anh của vi điều khiển họ 8051 thông qua bộ đếm 74HC393 để đạt được xung clock phù hợp cho ADC 0809.

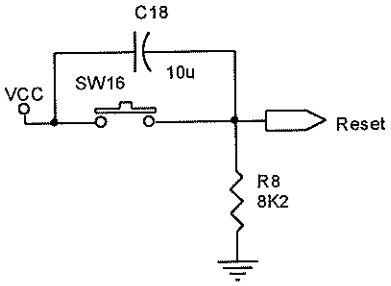


Clock vào lấy từ chân Xtal 2 (12 Mhz) của vi điều khiển 89C51 , ngõ ra lấy ở chân Q_D của bộ đếm để chia 16. Vậy tần số có thể đưa vào ADC là : 12 Mhz / 16 = 750 Khz. Ở đây, tuy tín hiệu vào ở dạng sóng sin nhưng do 74HC393 có độ nhạy cao nên hoàn toàn không cần biến đổi xung sin thành xung vuông.

2.2.4. Khối vi điều khiển :

2.2.4.1. Mạch reset hệ thống :

Để reset 89C51 qua phím ấn, nguồn phải được trạng thái khoảng 90% giá trị của nó và mạch dao động cũng phải đạt đến trạng thái có tỉ số tín hiệu trên nhiễu đủ tốt và chu kì dao mà nó tạo ra đủ chính xác. Khi đã đạt được các yêu cầu trên, tín hiệu ở chân reset cần phải ở mức cao ít nhất 2 chu kỳ máy thì hệ thống mới có thể được reset đúng. Theo datasheet của AT89C51, mức điện áp cao này là V_{IH1} = 3,5V. Sơ đồ mạch reset ở mức cao như sau :



Thời gian reset có thể chọn như sau :

$T_{reset} = T_{VccRise} + T_{OSC} + T_{2x\ ma.cycle} \approx T_{VccRise} + T_{OSC}$

$T_{VccRise}$: thời gian tăng của nguồn (10% đến 90 %),phụ thuộc nguồn cấp, khoảng 1ms đến 100ms.

T_{OSC} : thời gian để mạch dao động đạt trạng thái có thể reset, phụ thuộc đặc tính thạch anh và các tụ nối, khoảng 1ms đến 50ms.

$T_{2x\ ma.cycle}$: 2 chu kỳ máy của 89C51, dài 2 μs đối với thạch anh 12Mhz.

$$V_{reset} = Vcc^* \exp(\frac{T_{reset}}{R8 * C18}) = V_{IH1} = 3,5$$

$$\text{Nếu } T_{reset} = 30ms, R_8 = 8K2 \Rightarrow C_{18} = \frac{30.10^{-3}}{8,2.10^3.\ln(\frac{5}{3,5})} = 10,25\ \mu F$$

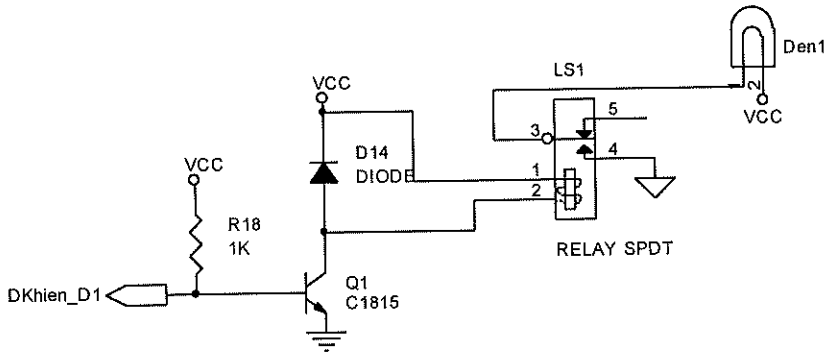
Chọn $C_{18} = 10\ \mu F$ và $R_8 = 8,2\ K\Omega$ là cặp giá trị thường được sử dụng.

Trong trường $T_{reset} = 150ms$, $R_8 = 8K2$

$$\Rightarrow C_{18} = \frac{150.10^{-3}}{8,2.10^3.\ln(\frac{5}{3,5})} = 35,8\ \mu F$$

2.2.4.2. Mạch điều khiển role :

Để đóng mở tải công suất, ta có thể dùng Role vì có đặc điểm là thiết kế đơn giản, dễ sử dụng. Sơ đồ mạch như sau :



Tín hiệu điều khiển đèn 1 Dkhien_D1 từ vi điều khiển đưa đến cực B của transitor thông qua R_{18} . Chọn Role 5V có điện trở cuộn dây 70 Ohm. Role này sẽ đóng khi điện áp của nó từ 3,75V – 5V, ngắt khi áp xuống 0,5V. Ta sẽ thiết kế để điện áp đóng role là 5V.

Chọn transitor C1815 :

$I_{CMAX} = 150mA$

$\beta = 100 - 700$

$V_{CESAT\ MAX} = 0.25V$

$V_{BESAT\ MAX} = 1V$

Để transitor bão hòa, ta cần điều kiện :

$I_{BS} = S\ I_{CS} / \beta_{min}$. (S > 1 : tượng trưng cho mức độ bão hòa) (1)

Để chắc chắn transitor bão hòa, ta tính cho trường hợp I_C càng lớn và I_B càng nhỏ càng tốt tương ứng với giá trị $V_{CESAT} = 0$, và $V_{BESAT\ MAX} = 1$.

Dòng qua role khi transitor bão hòa:

$$I_{Relay} = I_{CS} = \frac{5 - V_{CESAT}}{70} = \frac{5-0}{70} = 71\ mA\ < I_{CMAX}$$

Trong các datasheet, thường cho $I_C = 10\ I_B$ để transitor rơi vào trạng thái bão hòa. Do đó ta cho :

$$I_{BS} = \frac{I_{CS}}{10} = \frac{71}{10} = 7{,}1\ mA \Leftrightarrow I_{BS} = 10\frac{I_{CS}}{\beta_{min}} \Leftrightarrow S = 10$$

$$R_{18} = \frac{Vcc - V_{BESAT\ MAX}}{I_{BS}} = \frac{5-1}{7{,}1} = 560\ \Omega.$$

Với giá trị $R_{18} = 560\ \Omega$, khi mắc vào mạch thực tế, V_{CESAT} tăng lên (> 0), V_{BESAT} giảm xuống (< 1) khiến dòng I_{CS} giảm xuống (< 71 mA) và I_{BS} tăng lên (> 7,1 mA). Dù vậy, điều kiện (1) luôn thỏa.

Tuy nhiên, khi tắt role ta cần đảm bảo I_{OL} < 10mA (dòng ra mức thấp của chân vi điều khiển 89C51) $\Leftrightarrow R_{18} > \frac{V_{CC}}{10} = \frac{5}{10} = 0{,}5K\Omega$. Vậy chọn $R_{18} = 1K\Omega$ với :

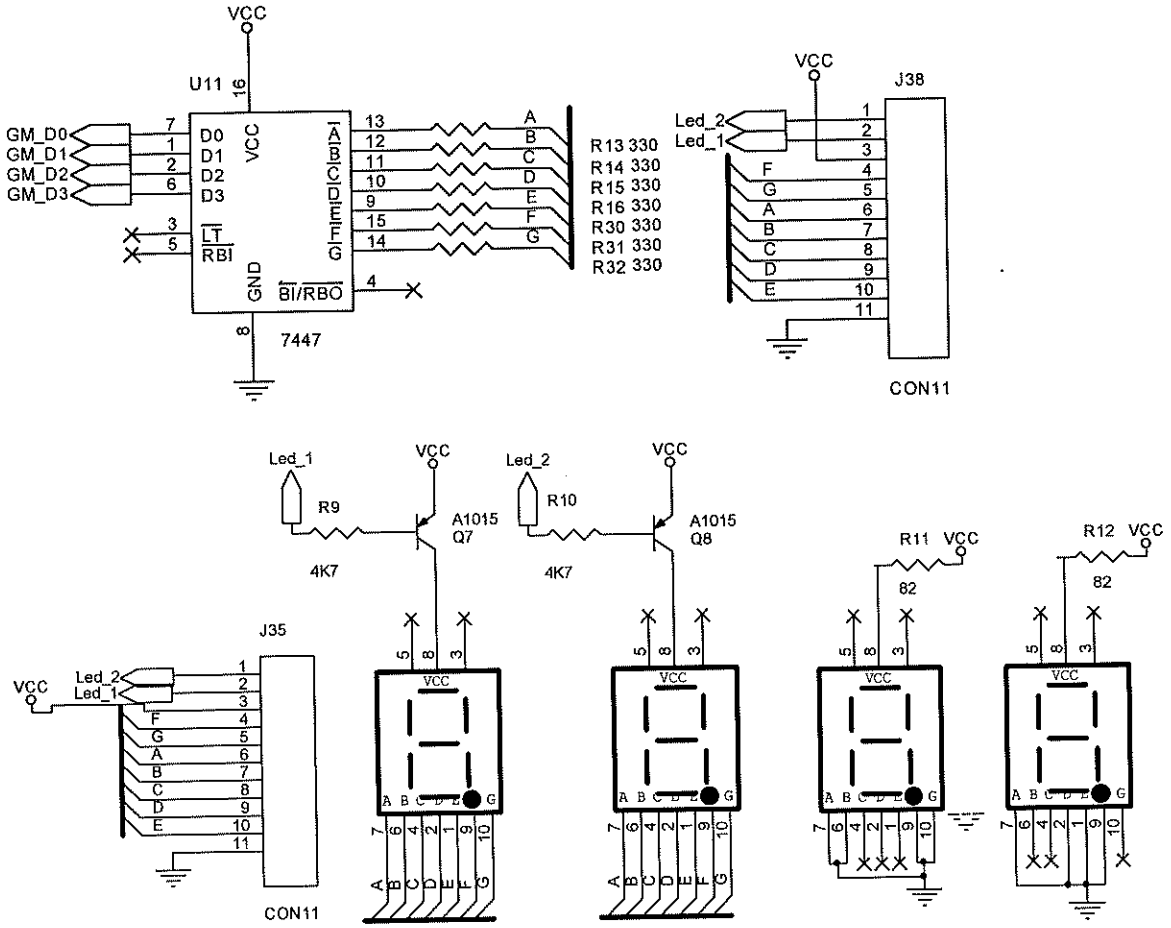
$$I_{BS} = \frac{Vcc - V_{BESAT\ MAX}}{R_{18}} = \frac{5-1}{1} = 4mA \text{ và } S = \frac{I_{BS}\beta_{min}}{I_{CS}} = \frac{4.100}{71} \approx 6$$

Do mạch thiết kế với các điện trở thanh gắn vào Port P2 của 89C51, giá trị điện trở thanh có thể chọn là $1K\Omega$ cho mỗi điện trở.

2.2.5. Khối hiển thị LED :

Để hiển thị nhiệt độ, ta cần 2 led sẽ hiển thị số từ 0 đến 99 có cả dấu thập phân, 1 led cho phần thập phân, 2 led cho hiển thị kí hiệu "°C". Tuy nhiên, do các port của 89C51 đã được dùng hết nên phần hiển thị thập phân sẽ không thực hiện bằng Led. Mạch hiển thị Led sẽ thể hiện nhiệt độ ở phòng 1. Có thể xem nhiệt độ 2 phòng (cả phần thập phân) trên giao diện phần mềm của PC . Để tiết kiệm port, IC giải mã BCD-Led 7 đoạn 74LS47 được sử dụng.

Sơ đồ mạch như sau :



IC giải mã đầu ra tích cực mức thấp, vì vậy các Led sử dụng phải là loại anode chung . Các led được gắn rời nên cần đầu nối J38 11 chân. Hai chân Led_1, Led_2 là chân chọn led, tín hiệu chọn lấy từ 2 chân của 89C51. Các chân chọn led được sử dụng vì phương pháp hiển thị quét led được chọn. So với phương pháp chốt,

phương pháp quét led sẽ đơn giản phần cứng hơn, phần mềm không quá khó và ư điểm là ít tiêu tốn công suất từ nguồn.

Điện áp rơi trên các đoạn của Led thường là 2V, dòng cung cấp tốt nhất cho mỗi đoạn khoảng 10 mA. Để hiện thị ký hiệu nhiệt độ, mỗi led cần thấp sáng 4 đoạn tương đương với dòng tổng qua 4 led là 40 mA. Như vậy áp rơi trên các điện trở R_{11}, R_{12} sẽ là : Vcc-2 = 5-2 = 3V và dòng qua nó là 40 mA. Từ đó suy ra giá trị các điện trở $R_{11} = R_{12} = \frac{3}{40} = 75\ \Omega$. Vậy ta chọn $R_{11} = R_{12} = 82\ \Omega$ là giá trị tiêu chuẩn.

Để tính các trở R_9 và R_{13} cho mạch chọn led, ta chọn transitor A1015 :

$I_{CMAX} = -150mA$

$\beta = 70 - 400$

$V_{CESAT\ MAX} = -0,3\ V$

$V_{BESAT\ MAX} = -1,1\ V$

Tương tự mạch điều khiển rơle, ta tính khi $V_{CESAT} = 0$ và $V_{BESAT\ MAX} = -1.1$ V. Áp rơi trên 1 đoạn Led là khoảng 2 V, dòng cấp tốt nhất là 10mA, suy ra áp trên R_{13} là 5-2 = 3 V. Giá trị điện trở 330 Ω được chọn.

Tương tự mạch điều khiển role, $I_{CS} = -10$ mA , để $I_{CS} = 10\ I_{BS}$ thì $I_{BS} = -1$ mA

(hệ số bão hòa $S = \frac{I_{BS}\beta_{min}}{I_{CS}} = \frac{1.70}{10} = 7$)

$$\Rightarrow R_{18} = \frac{Vcc + V_{BESAT\ MAX}}{-I_{BS}} = \frac{5 - 1,1}{-1} = 3,9K\Omega$$

Ta có thể tăng R_{18} lên 4,7 KΩ để giảm dòng I_{BS} 0,8 mA và S = 5,6.

2.2.6. Khối điều khiển thiết bị từ xa :

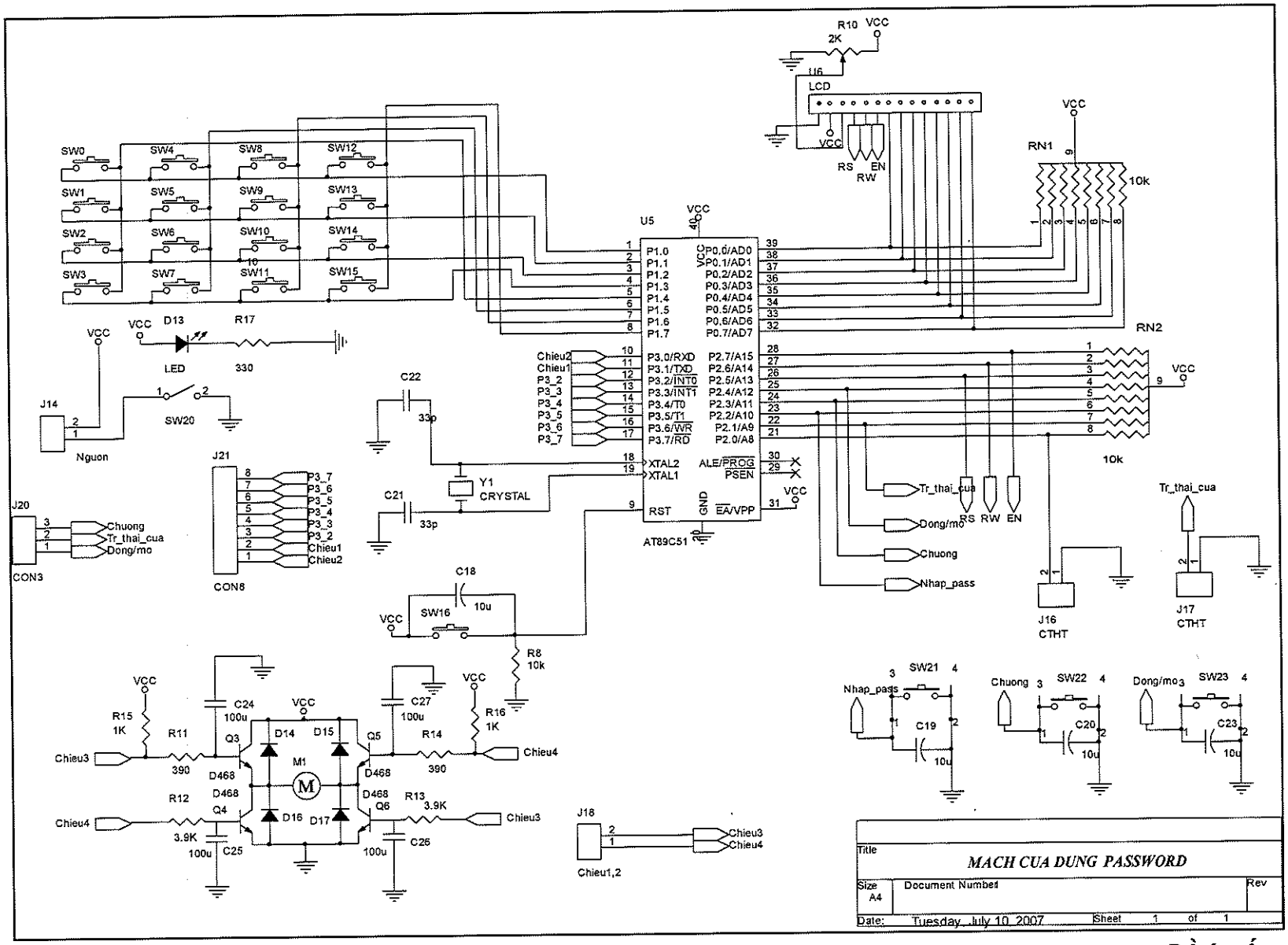
Gồm khối thu dùng SC9149 (nằm trên mạch) và khối phát dùng SC9148 (dạng remote cầm tay)

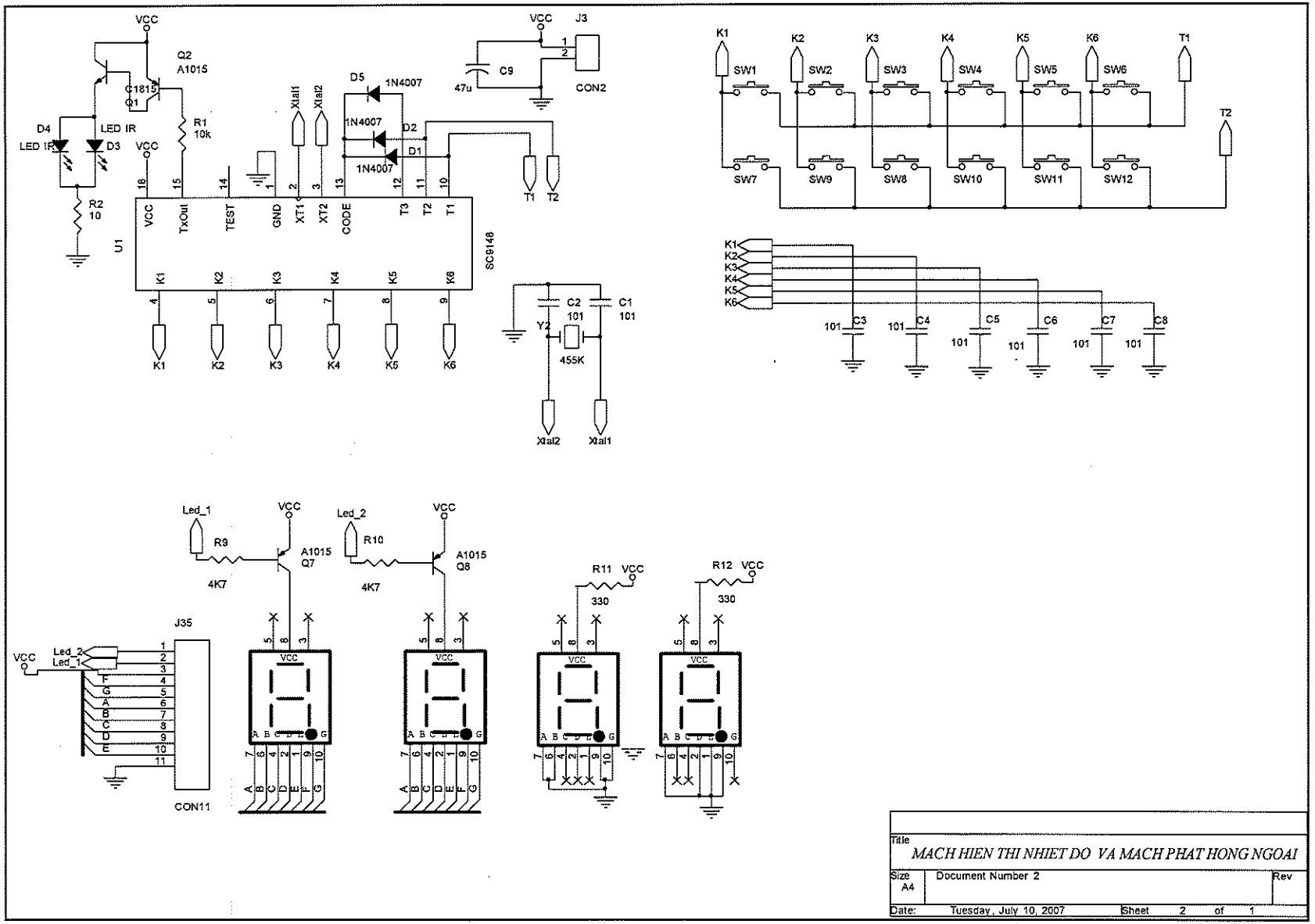
Nguyên lý hoạt động và sơ đồ ứng dụng đã được đề cập ở phần 2 mục 5

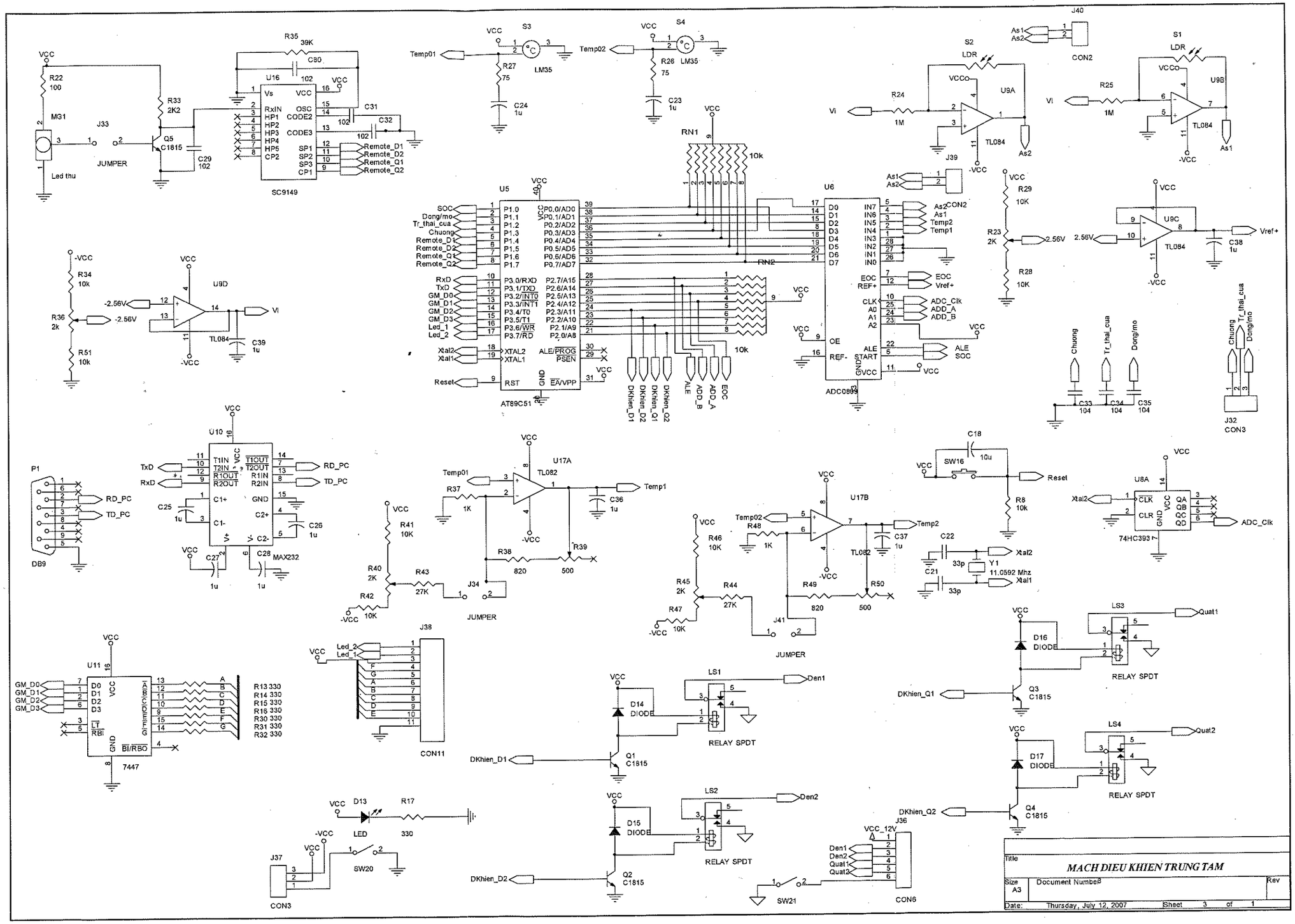
2.2.7. Khối giao tiếp với máy vi tính :

Nguyên tắc hoạt động và sơ đồ thiết kế đã được đề cập ở phần 2 mục 6

3. Sơ đồ nguyên lý tổng thể hệ thống:


R10 VCC
2K
U16
LCD
VCC
RS EN
RW
RN1
10k
SW0 SW4 SW8 SW12
SW1 SW5 SW9 SW13
SW2 SW6 SW10 SW14
SW3 SW7 SW11 SW15
U5
VCC
P1.0 P1.1 P1.2 P1.3 P1.4 P1.5 P1.6 P1.7
P0.0/AD0 P0.1/AD1 P0.2/AD2 P0.3/AD3 P0.4/AD4 P0.5/AD5 P0.6/AD6 P0.7/AD7
RN2
Chieu2 Chieu1 P3_2 P3_3 P3_4 P3_5 P3_6 P3_7
P3.0/RXD P3.1/TXD P3.2/INT0 P3.3/INT1 P3.4/T0 P3.5/T1 P3.6/WR P3.7/RD
P2.7/A15 P2.6/A14 P2.5/A13 P2.4/A12 P2.3/A11 P2.2/A10 P2.1/A9 P2.0/A8
VCC D13 R17
LED 330
J14
SW20
Nguon
C22 33p
C21 33p
Y1 CRYSTAL
XTAL2 XTAL1 RST
ALE/PROG PSEN EA/VPP GND
AT89C51
J21
CON8
J20
CON3
Chuong
Tr_thai_cua
Dong/mo
Nhap_pass
J16 CTHT
J17 CTHT
C18 10u
SW16
R8 10k
R15 1K
R16 1K
C24 100u
C27 100u
R11 390
R14 390
Q3 Q5
D14 D15 D468
M1
R12 3.9K
R13 3.9K
Q4 Q6
D16 D17
C25 100u
C26 100u
Chieu3
Chieu4
J18
Chieu1,2
SW21 SW22 SW23
C19 C20 C23 10u
Title
MACH CUA DUNG PASSWORD
Size A4
Document Number
Rev
Date: Tuesday, July 10, 2007
Sheet 1 of 1


VCC
Q2
A1015
C1815
Q1
D4
LED IR
LED IR
D3
R1
10k
R2
10
VCC
Xtal1
Xtal2
D5
1N4007
1N4007
D2
1N4007
D1
VCC
J3
C9
47u
CON2
T1
T2
U1
VCC
TxOut
TEST
GND
XT1
XT2
CODE
T3
T2
T1
SC9148
K1
K2
K3
K4
K5
K6
C2
101
C1
101
Y2
455K
Xtal2
Xtal1
K1
K2
K3
K4
K5
K6
T1
T2
SW1
SW2
SW3
SW4
SW5
SW6
SW7
SW9
SW8
SW10
SW11
SW12
K1
K2
K3
K4
K5
K6
101
C3
101
C4
C5
101
C6
101
C7
101
C8
101
Led_1
VCC
R9
4K7
A1015
Q7
Led_2
VCC
R10
4K7
A1015
Q8
R11
VCC
330
R12
VCC
330
J35
VCC
Led_2
Led_1
F
G
A
B
C
D
E
CON11
VCC
A B C D E F G
A B C D E F G
A B C D E F G
A B C D E F G
Title
MACH HIEN THI NHIET DO VA MACH PHAT HONG NGOAI
Size
A4
Document Number 2
Rev
Date:
Tuesday, July 10, 2007
Sheet
2
of
1


SC9149
AT89C51
ADC0809
MAX232
7447
CON2
CON3
CON11
CON6
JUMPER
RELAY SPDT
DIODE
LM35
TL084
TL082
74HC393
MACH DIEU KHIEN TRUNG TAM
Document Number
Thursday, July 12, 2007
Sheet 3 of 1

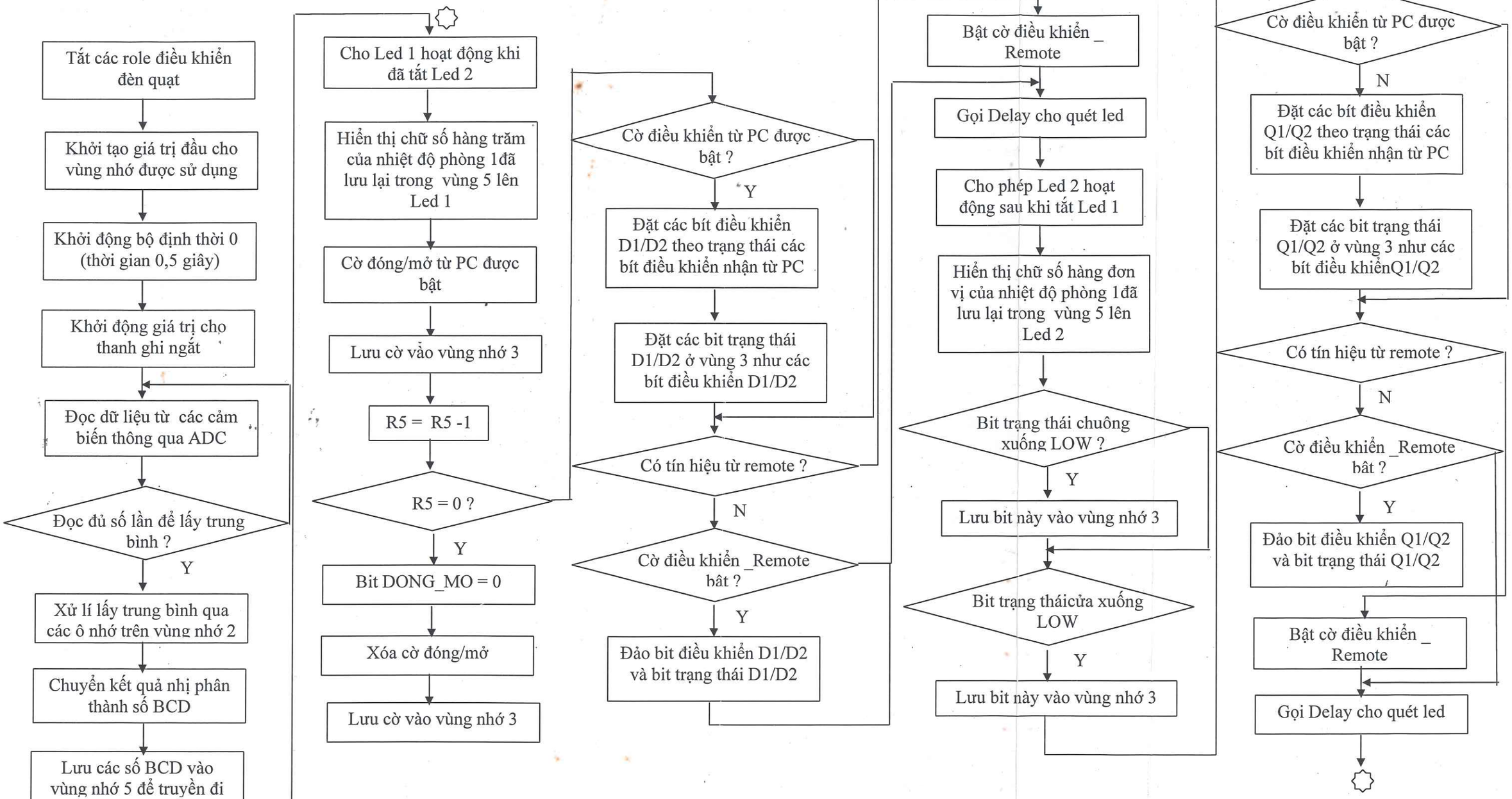
4. Lưu đồ thuật toán cho phần mềm:

4.1. Lưu đồ thuật toán của chương trình mạch điều khiển trung tâm :

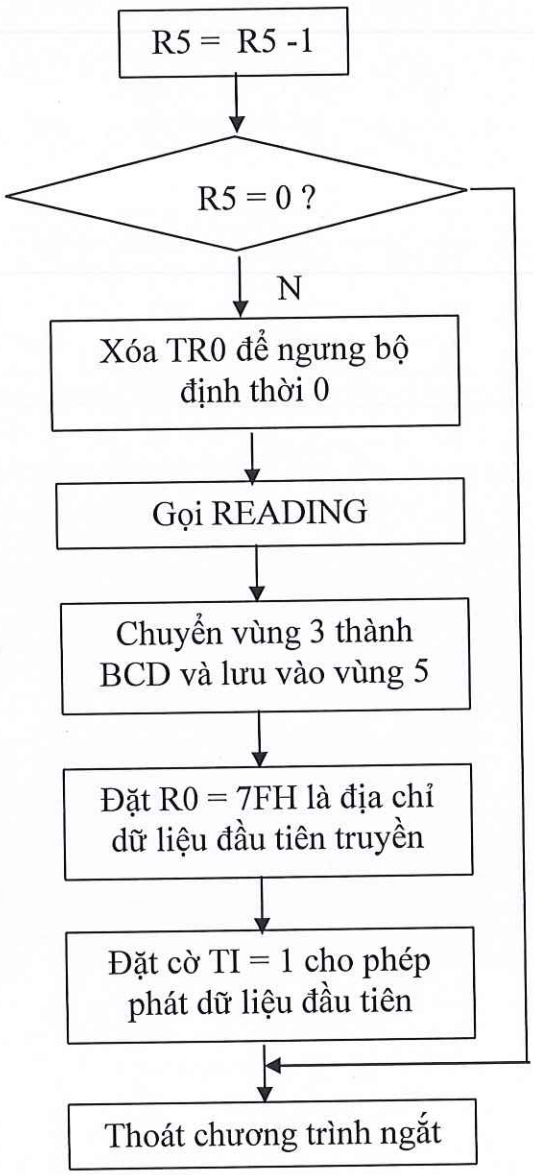
Vùng nhớ ram được sử dụng :

- ✓ Vùng 1 : Bank 0, chứa các biến R0, R1 làm con trỏ, R3 cho việc qui định thời gian định thời, R5 qui định độ rộng xung đóng mở, R4 qui định số lần lấy trung bình
- ✓ Vùng 2 : Thanh ghi 18H, 19H cho việc xử lí trung bình, thanh ghi 2FH cho cờ điều khiển từ Remote
- ✓ Vùng 3 : thanh ghi 20H cho trạng thái thiết bị đèn 1 (D1), đèn 2(D2), quạt 1(Q1), quạt 2 (Q2), trạng thái chuông , trạng thái cửa, bit báo đã nhận lệnh đóng mở từ PC
- ✓ Vùng 4 : thanh ghi 23H cho việc lưu dữ liệu nhận từ PC
- ✓ Vùng 5 : thanh ghi 7FH đến thanh ghi 6DH cho dữ liệu truyền lên PC

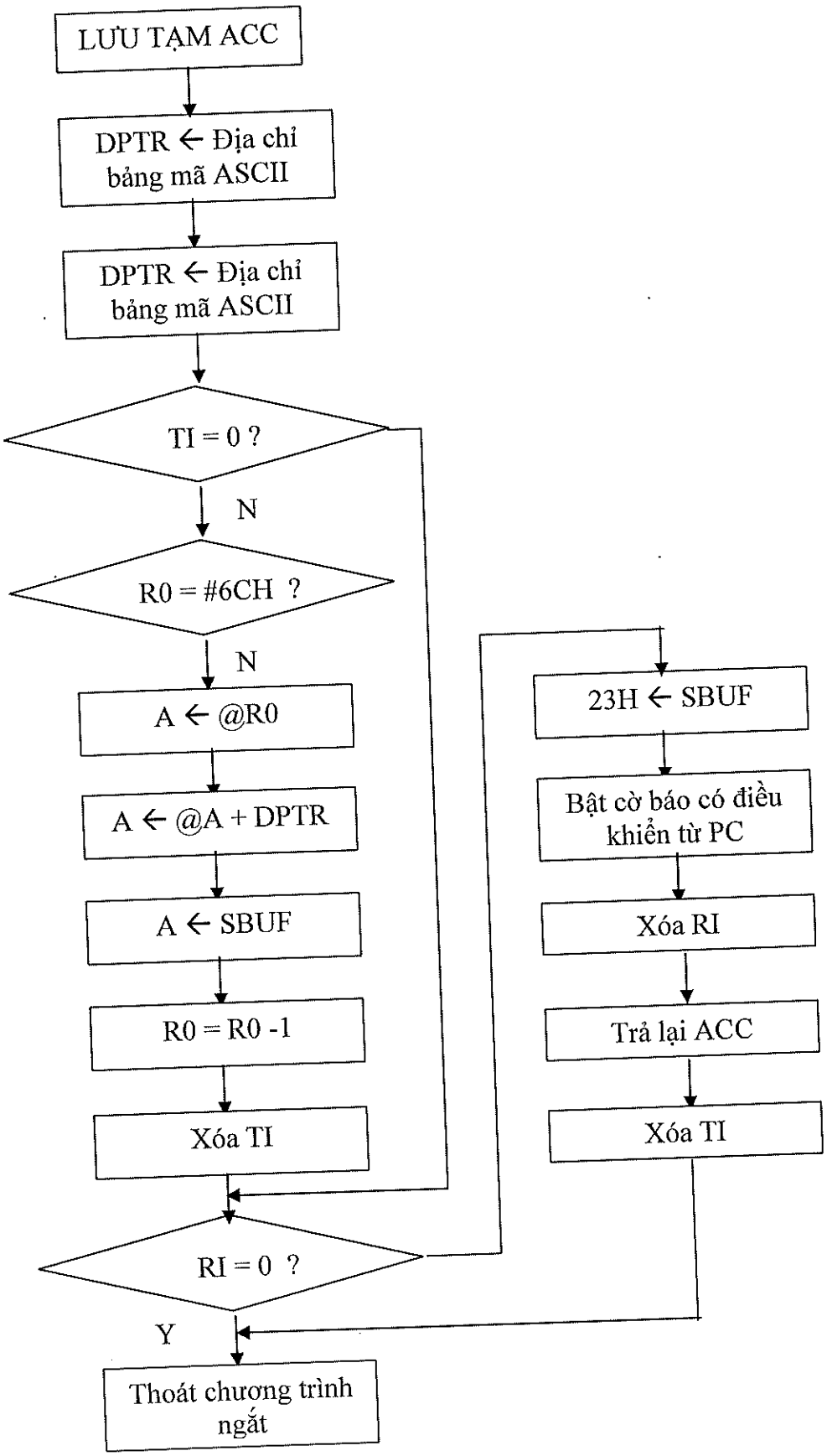
CHƯƠNG TRÌNH ĐIỀU KHIỂN TRUNG TÂM



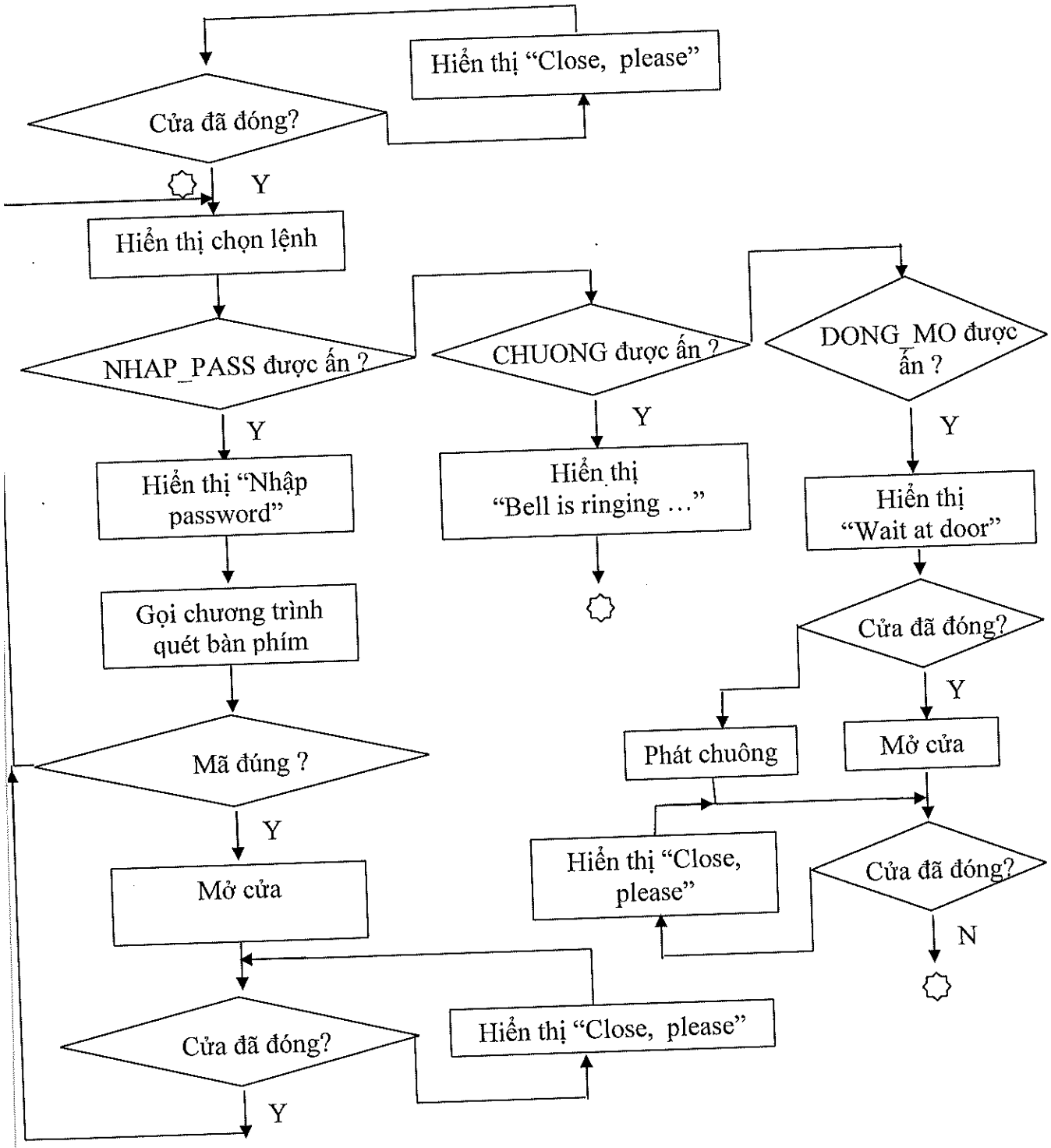
CHƯƠNG TRÌNH CON PHỤC VỤ NGẮT ĐỊNH THỜI



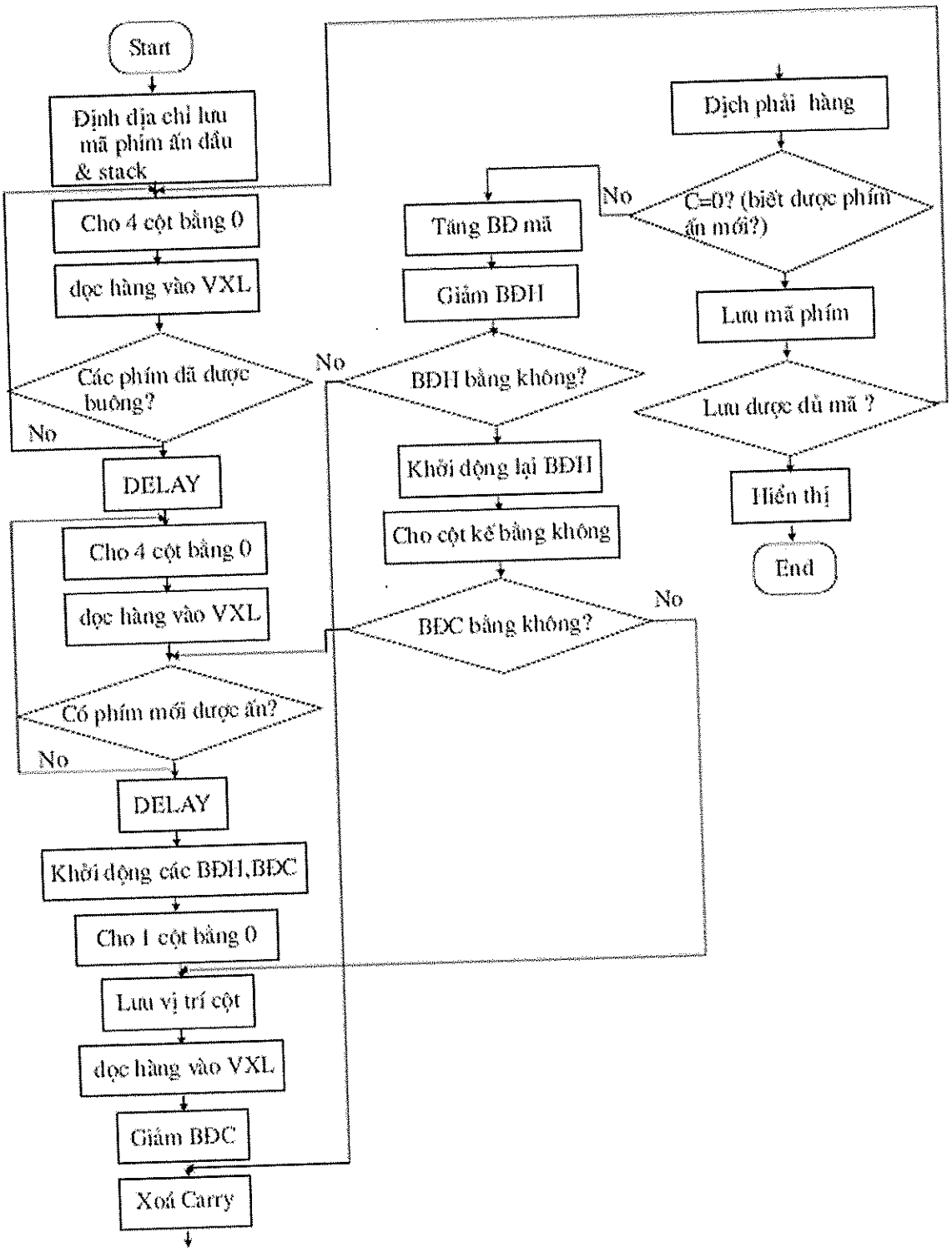
CHƯƠNG TRÌNH CON PHỤC VỤ NGẮT NỐI TIẾP



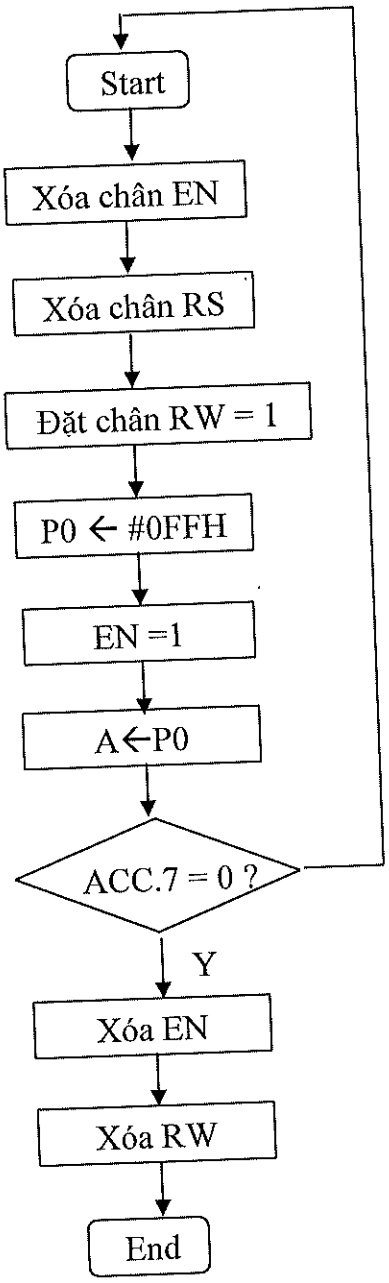
4.2. Lưu đồ thuật toán của chương trình mạch điều khiển cửa :



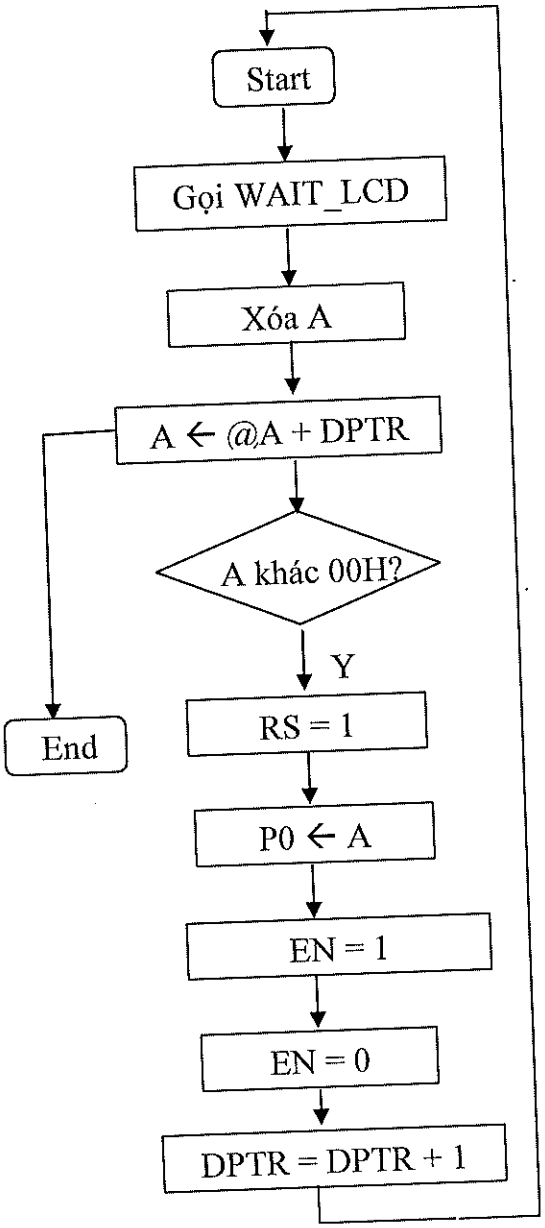
Lưu đồ thuật toán chương trình con quét bàn phím :



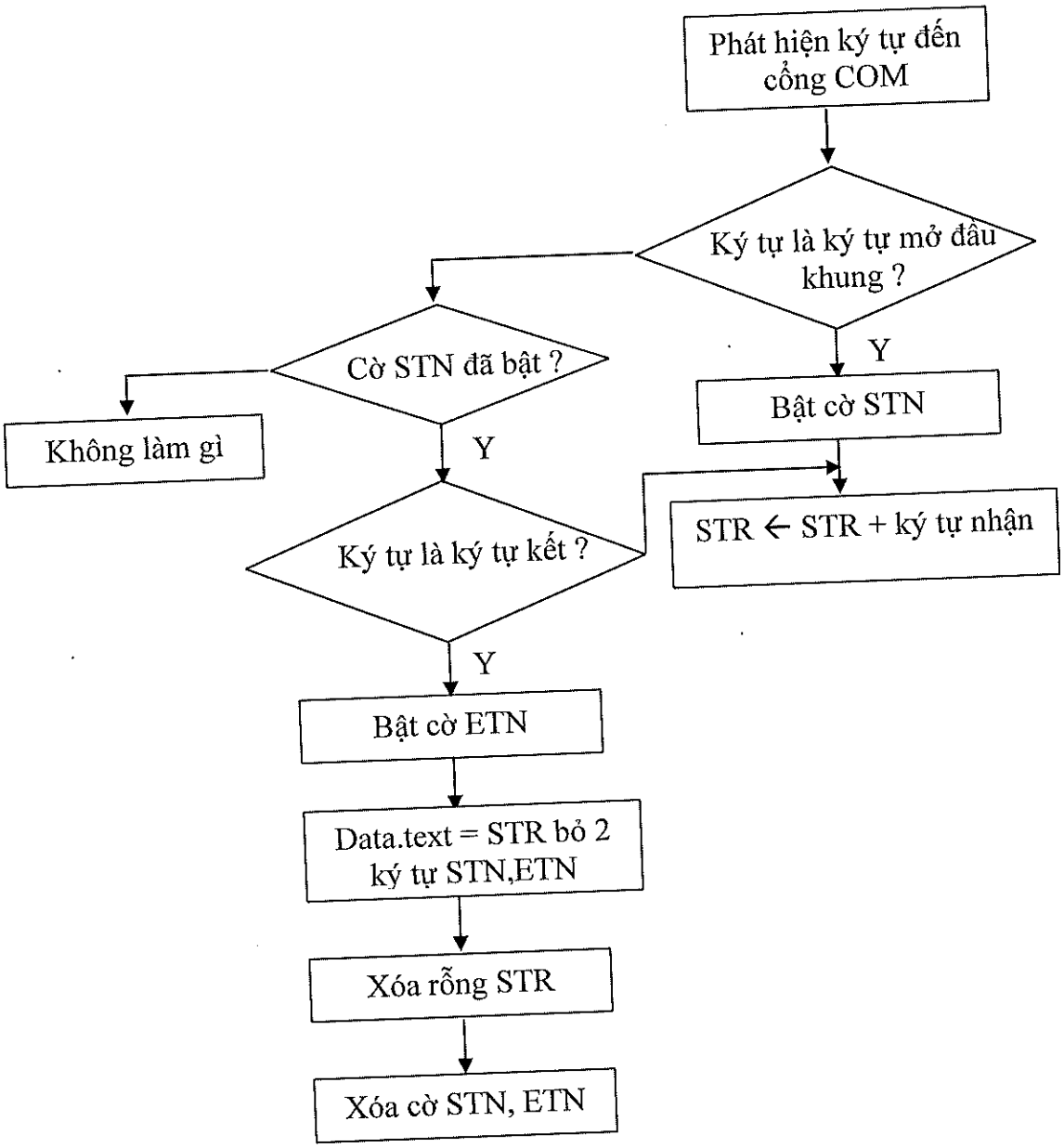
Lưu đồ thuật toán chương trình con WAIT_LCD :

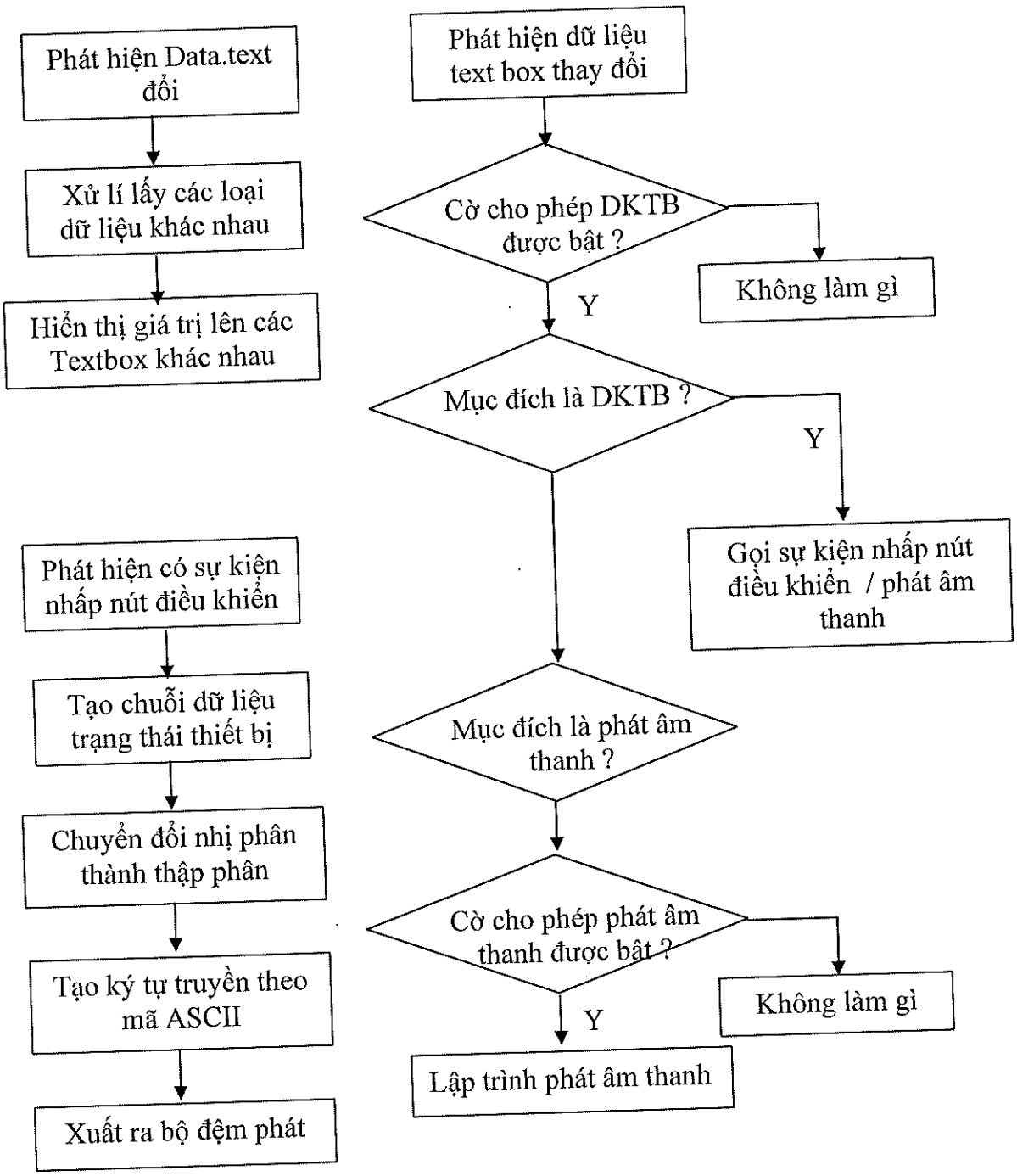


Lưu đồ thuật toán chương trình con xuất chuỗi ra LCD :



4.3. Lưu đồ thuật toán của phần mềm trên máy tính:




Phát hiện Data.text đổi
Xử lí lấy các loại dữ liệu khác nhau
Hiển thị giá trị lên các Textbox khác nhau
Phát hiện dữ liệu text box thay đổi
Cờ cho phép DKTB được bật ?
Không làm gì
Y
Mục đích là DKTB ?
Y
Gọi sự kiện nhấp nút điều khiển / phát âm thanh
Phát hiện có sự kiện nhấp nút điều khiển
Tạo chuỗi dữ liệu trạng thái thiết bị
Chuyển đổi nhị phân thành thập phân
Tạo ký tự truyền theo mã ASCII
Xuất ra bộ đệm phát
Mục đích là phát âm thanh ?
Cờ cho phép phát âm thanh được bật ?
Không làm gì
Y
Lập trình phát âm thanh

PHẦN 4 : KẾT LUẬN

Sau 3 tháng thực hiện với nổ lực và cố gắng của bản thân cùng với sự hướng dẫn tận tình của cô Ths.Nguyễn Thị Ngọc Anh, tập luận văn đã hoàn thành đúng thời hạn. Theo yêu cầu đặt ra là thiết kế ngôi nhà thông minh, người thiết kế đã hoàn thành một hệ thống nhỏ điều khiển, hiển thị ánh sáng, nhiệt độ, các thiết bị và đảm bảo an ninh cho ngôi nhà.

Hệ thống có một số khuyết điểm :

- ✓ Không thể thay đổi độ sáng đèn liên tục
- ✓ Phần cảm biến nhiệt không thể lưu giữ nhiệt độ tại thời điểm cúp điện
- ✓ Mạch điều khiển cửa bằng password không thể đặt password mới khi đang hoạt động

Hướng phát triển để tài :

- ✓ Cần lắp đặt thêm các IC nhớ , thiết kế các mạch điều chỉnh độ sáng đèn dùng linh kiện triac
- ✓ Hướng đến thiết kế hệ thống lớn theo chuẩn X10 hoặc INSTEON như đã đề cập.

PHỤ LỤC

Bảng mã X10 :

X10 Code Chart			
X10 Code	**Hex Value**	**X10 Code**	**Hex Value**
A	46	8	5A
B	4E	9	4E
C	42	10	5E
D	4A	11	46
E	41	12	56
F	49	13	40
G	45	14	50
H	4D	15	48
I	47	16	58
J	4F	All Units Off	41
K	43	All Lights On	43
L	4B	On	45
M	40	Off	47
N	48	Dim	49
O	44	Bright	4B
P	4C	All Lights Off	4D
1	4C	Extended Code	4F
2	5C	Hail Request	51
3	44	Pre-Set Dim High	57
4	54	Pre-Set Dim Low	55
5	42	Extended Data (analog)	59
6	52	Status = on	5B
7	4A	Status = off	5D
		Status Request	5F

Chương trình điều khiển cửa :

```
ORG 0000H
DA_MO_HET       BIT P2.0
DA_DONG_SAT     BIT P2.1
NHAP_PASS       BIT P2.2
CHUONG    BIT P2.3
DONG_MO   BIT P2.4
CHIEU1    BIT P3.1
CHIEU2    BIT P3.0

EN    BIT P2.7
RS    BIT P2.5
RW    BIT P2.6
      MOV SP,#2FH
      MOV  20H,#00H
      CLR  CHIEU1
      CLR  CHIEU2
      CLR  CHIEU1
      CLR  CHIEU2
      LCALL INIT_LCD
      LCALL       CLEAR_LCD
KTRA_CUA:
      LCALL       KTRA_CUA_DONG
      MOV  R5,#30                    ;SO LAN HIEN BELL IS RINGING
CHONLENH:
      LCALL CLEAR_LCD
      MOV DPTR, #CHON_NHAPPASS                ;HIEN THI CHON
NHAP PASS
      LCALL STRINGOUT

      LCALL WAIT_LCD
      CLR RS
    MOV P0,#0C0h                     ;DINH CON TRO
    SETB EN
      CLR EN

      MOV DPTR, #CHON_ANCHUONG                ;HIEN THI CHON AN
 CHUONG
      LCALL       STRINGOUT
START :
      SETB DA_DONG_SAT
      SETB DA_MO_HET
```

```
        SETB NHAP_PASS
        SETB CHUONG
        SETB DONG_MO


QUET_NUT_DK :
QUET_NHAP_PASS :
BUONG_NHAP_PASS:
        MOV  C,NHAP_PASS                    ;KIEM TRA PHIM BUONG CHUA
        JNC   BUONG_NHAP_PASS
AN_NHAP_PASS:
        MOV  C,NHAP_PASS                    ;BAT DAU KIEM TRA PHIM
NHAP_PASS
        JC      QUET_CHUONG
        LCALL       DELAY
        JC      QUET_CHUONG
        LCALL CLEAR_LCD
        MOV DPTR, #NHAPPASS                 ;HIEN THI NHAP PASS
        LCALL STRINGOUT

        LCALL WAIT_LCD
        CLR RS
     MOV P0,#0C5h                           ;DINH CON TRO DE CHUAN BI HIEN THI
  PASS
     SETB EN
        CLR EN

        MOV R1,#15H
        LJMP PHIMBUONG                      ;BAT DAU QUET BAN PHIM


  QUET_CHUONG:
  BUONG_CHUONG :
        MOV  C, CHUONG
        JNC   BUONG_CHUONG
  AN_CHUONG      :
        MOV  C, CHUONG
        JC      QUET_DONG_MO
        LCALL       DELAY
        JC      QUET_DONG_MO
        LCALL CLEAR_LCD
   BELL_RINGING:
        LCALL WAIT_LCD
```

```
    CLR RS
  MOV P0,#080h                    ;DINH CON TRO
  SETB EN
     CLR EN

     MOV DPTR, #WAIT_A_MINUTE                    ;HIEN THI CHO
     LCALL STRINGOUT
     LCALL DELAY
     LCALL DELAY
     DJNZ R5,BELL_RINGING
     MOV  R5, #30
     LJMP CHONLENH

QUET_DONG_MO :
BUONG_DONG_MO:
       MOV  C, DONG_MO
       JNC   BUONG_DONG_MO

AN_DONG_MO:
       MOV  C, DONG_MO
       JC      TAM7
       LCALL DELAY
       JC      TAM7
       LCALL CLEAR_LCD
       MOV DPTR, #AT_DOOR              ;HIEN THI DOI O CUA
       LCALL STRINGOUT



       LJMP XU_LI_DONG_MO
 TAM7:
       LCALL KTRA_CUA_DONG
       LJMP CHONLENH
 XU_LI_DONG_MO:
       JNB   DA_MO_HET,TAM5
       JB      DA_DONG_SAT,QUET_NUT_DK
       CLR   CHIEU1
       CLR   CHIEU2
       LCALL DELAY
       LCALL DELAY
       SETB CHIEU1
       JB      DA_MO_HET,$
       LCALL       DELAY
```

```
        JB      DA_MO_HET,$
        CLR   CHIEU1

        LCALL DELAY
        LCALL DELAY
        LJMP KTRA_CUA

TAM5:       JNB   DA_DONG_SAT,TAM6
        CLR   CHIEU1
        CLR   CHIEU2
        LCALL DELAY
        LCALL DELAY
        SETB CHIEU2
        JB      DA_DONG_SAT,$
        LCALL       DELAY
        JB      DA_DONG_SAT,$

        CLR   CHIEU2
        LCALL DELAY
        LCALL DELAY
        LJMP KTRA_CUA
 TAM6:
        LJMP KTRA_CUA

 KTRA_CUA_DONG :
        SETB 00H
 AGAIN:
        JB      DA_DONG_SAT,HTHI_GOI_DONG_CUA
        RET


  HTHI_GOI_DONG_CUA:

        JNB   00H, BUONG_DONG_MO2
        CLR   00H
        LCALL CLEAR_LCD
        MOV DPTR, #GOI_DONG_CUA                 ;HIEN THI CHON NHAP PASS
        LCALL STRINGOUT

   BUONG_DONG_MO2:
        MOV  C, DONG_MO
        JNC   BUONG_DONG_MO2
```

```
AN_DONG_MO2:
        MOV  C, DONG_MO
        JC      AGAIN
        LCALL DELAY
        MOV  C, DONG_MO
        JC      AGAIN
        LCALL       DONG_CUA
        SJMP  AGAIN


 DONG_CUA:
        CLR   CHIEU1
        CLR   CHIEU2
        LCALL DELAY
        LCALL DELAY
        SETB CHIEU2
        JB      DA_DONG_SAT,$
        LCALL       DELAY
        JB      DA_DONG_SAT,$

        CLR   CHIEU2
        LCALL DELAY
        LCALL DELAY
  RET


  MO_CUA:
        CLR   CHIEU1
        CLR   CHIEU2
        LCALL DELAY
        LCALL DELAY
        SETB CHIEU1
        JB      DA_MO_HET,$
        LCALL       DELAY
        JB      DA_MO_HET,$
        CLR   CHIEU1
        LCALL DELAY
        LCALL DELAY
    RET
```

```
WAIT_LCD:
    CLR EN ;Start LCD command Hienthi
    CLR RS ;It's a command
    SETB RW ;It's a read command
    MOV P0,#0FFh ;Set all pins to FF initially
    SETB EN ;Clock out command to LCD
    MOV A,P0 ;Read the return value
    JB ACC.7,WAIT_LCD ;If flag high, LCD still busy
    CLR EN ;Finish the command
    CLR RW ;Turn off RW for future commands
    RET
INIT_LCD:
       LCALL WAIT_LCD
     CLR RS
     MOV P0,#38h
     SETB EN
     CLR EN

       LCALL WAIT_LCD
     CLR RS
     MOV P0,#0Fh
     SETB EN
       CLR EN

       LCALL WAIT_LCD
       CLR RS
     MOV P0,#06h
     SETB EN
       CLR EN
       RET

  CLEAR_LCD:
       LCALL WAIT_LCD
       CLR RS
     MOV P0,#01h
     SETB EN
       CLR EN
     RET

   WRITE_TEXT:
         SETB RS
      MOV P0,A
      SETB EN
```

```
    CLR EN
    RET
STRINGOUT:
       LCALL WAIT_LCD
       CLR A
       MOVC A, @A+DPTR
       CJNE A, #00H, XUAT
       SJMP THOATSTRING
XUAT:

       LCALL WRITE_TEXT
       INC DPTR
       SJMP STRINGOUT
THOATSTRING:
       RET
TAM3 :
       SJMP  TAM3
PHIMBUONG: MOV P1,#0FH
           MOV A,P1
           ANL A,#0FH
           CJNE A,#0FH,PHIMBUONG

 PHIMAN:
           MOV P1,#0FH
           MOV A,P1
           CJNE A,#0FH,KTRA
          SJMP PHIMAN
 KTRA  :
          LCALL DELAY
          MOV R0,#0
          MOV R6,#4
          MOV R7,#4
          MOV P1,#0EFH

  QCOT :     MOV A,P1
          PUSH ACC
          DEC R7
   QHANG:
          CLR C
                ANL A,#0FH
          RRC A
          JNC LMA
          INC R0
```

```
        DJNZ R6,QHANG
        MOV R6,#4
        POP ACC
        RL A
        MOV P1,A
        CJNE R7,#0,QCOT
        LJMP PHIMAN
LMA  :
              MOV A,R0
              MOV @R1,A
              LCALL WAIT_LCD

              MOV A,@R1
              MOV DPTR,#ASCII
              MOVC A,@A+DPTR

              LCALL WRITE_TEXT

              DEC R1
              CJNE R1,#0FH,PHIMBUONG

 KIEMTRAMA:
              MOV R0,#15H        ; tro gian tiep den vung nho da luu phim an
        MOV R1,#00H  ;bien de dem so ky tu dung voi ma
              MOV DPTR, #PASSWORD
 REP   :
              CLR A
              MOVC A,@A+DPTR
         XRL A, @R0
              CJNE A,#00H,TAM
              INC R1
  TAM  :
        DEC R0
              INC DPTR
        CJNE R0,#0FH,REP

              CJNE R1,#06H, NHAPSAI ; nhap dung khi dung ca 6 ky tu
              SJMP NHAPDUNG

   NHAPDUNG :
              LCALL CLEAR_LCD
              MOV DPTR, #WELCOME
              LCALL STRINGOUT
```

```
            LCALL       MO_CUA
            LCALL       KTRA_CUA_DONG
            LJMP CHONLENH
NHAPSAI:
            LCALL CLEAR_LCD
            ;MOV DPTR, #NHAPLAI
            ;LCALL STRINGOUT
            LJMP CHONLENH

DELAY:
        MOV R7,#200
        MOV       R6,#80

LOOP:
        DJNZ R7,$
        MOV       R7,#200
        DJNZ R6,LOOP

RET
ASCII : DB '0123456789ABCDEF'
PASSWORD :              DB 00H,02H,04H,06H,08H,0AH
GOI_DONG_CUA :DB 'CLOSE, PLEASE',00H
NHAPPASS :              DB 'NHAP PASSWORD:',00H
CHON_NHAPPASS:      DB '1. NHAP PASSWORD',00H
CHON_ANCHUONG:      DB '2. AN CHUONG', 00H
WAIT_A_MINUTE :     DB 'BELL IS RINGING....',00H
AT_DOOR:          DB 'WAIT AT DOOR',00H
WELCOME :               DB 'WELCOME HOME!',00H
NHAPLAI :         DB 'NHAP LAI PASSWORD:',00H
END
```

Chương trình cho mạch điều khiển trung tâm :

```
SOC   BIT P1.0
      ALE   BIT P2.7
      ADDA        BIT P2.5
      ADDB        BIT P2.6
      EOC   BIT P2.4

      D0     BIT P3.2
      D1     BIT P3.3
      D2     BIT P3.4
      D3     BIT P3.5
      LED1 BIT P3.6
      LED2 BIT P3.7

      DEN1 BIT    P2.3
      DEN2 BIT    P2.2
      QUAT1 BIT  P2.1
      QUAT2       BIT    P2.0

      REMOTE_D1        BIT    P1.4
      REMOTE_D2        BIT    P1.5
      REMOTE_Q1        BIT    P1.6
      REMOTE_Q2        BIT    P1.7

      CO_DKHIEN_TU_PC      BIT 7BH
      CO_DONG_MO_CUA_TU_PC   BIT 1CH
      DONG_MO  BIT    P1.1
      CHUONG    BIT    P1.3
      TR_THAI_CUA BIT        P1.2
ORG 0000H
      MOV SP,#3FH
      LJMP MAIN

ORG 000BH
      LJMP TIMER_HALF_S
ORG 0023H
      LJMP SERIAL


ORG 0030H
MAIN:                                          ;CHUONG TRINH CHINH
CAMBIEN:
```

```
        MOV R0, #1FH
        CLR  DEN1
        CLR  DEN2
        CLR  QUAT1
        CLR  QUAT2
KHOIDONG1:
        MOV @R0, #00H
        DEC  R0
        CJNE R0,#08H,KHOIDONG1


      . MOV R0, #7EH
 KHOIDONG2:
        MOV @R0, #00H
        DEC  R0
        CJNE R0,#5FH,KHOIDONG2
        ;MOV        71H, #0
        ;MOV 70H,#1
        ;MOV 6FH,#1
      . MOV 7FH,#0AH                    ;LUU STN VAO 7FH
        MOV 6DH,#0BH                    ;LUU  ETN VAO 6DH
 THIETBI:
        ;CLR  DEN1
        ;CLR  DEN2
        ;CLR  QUAT1
        ;CLR  QUAT2
        SETB REMOTE_D1
        SETB REMOTE_D2
        SETB TR_THAI_CUA
        SETB CHUONG
        MOV 2FH, #00H
        MOV 20H,#00H

        MOV R5,#22                      ;THOI GIAN BIT DONG_MO
        MOV R4,#5                       ;TB
        MOV R3,#17                      ;TANG TIMER
        MOV TMOD, #01H                        ;KHOI DONG BO DINH THOI
  0
        MOV TH0, #80H
        MOV TL0, #01H
        SETB TR0
        MOV SCON,#50H
        MOV TMOD, #20H
```

```
MOV TH1, #0FDH
SETB TR1
SETB EA
SETB ES
SETB ET0

;SETB D0
;CLR D1
;CLR D2
;SETB D3
;LCALL WAIT


LCALL READING
LCALL HIENTHI

;CLR ADDA
;CLR ADDB
;LCALL READING
;SETB ADDA
;SETB ADDB
;LCALL READING
;SETB ADDB
;CLR ADDA
;LCALL READING


LJMP CAMBIEN

READING:
    PUSH ACC

    CLR  ALE                    ;DOC CAM BIEN 1
    CLR  SOC
    CLR  ADDB
    SETB ADDA
    LCALL DOC_AD
    MOV  R0,#18H
    LCALL CONG_DON

    CLR  ALE                    ;DOC CAM BIEN 2
    CLR  SOC
    CLR ADDA
```

```
CLR ADDB
LCALL DOC_AD
MOV R0,#1AH
LCALL CONG_DON

CLR  ALE                    ;DOC CAM BIEN 3
CLR  SOC
SETB ADDA
SETB ADDB
LCALL DOC_AD
MOV R0,#1CH
LCALL CONG_DON

CLR  ALE                    ;DOC CAM BIEN 4
CLR  SOC
SETB ADDB
CLR ADDA
LCALL DOC_AD
MOV R0,#1EH
LCALL CONG_DON

DJNZ R4, THOAT_READING

MOV R0,#18H                 ;LAM TRON CBND1
LCALL     LAM_TRON
MOV R0,#1AH                 ;LAM TRON CBND2
LCALL     LAM_TRON
MOV R0,#1CH                 ;LAM TRON CB3
LCALL     LAM_TRON
MOV R0,#1EH                 ;LAM TRON CB4
LCALL     LAM_TRON

MOV R0,#18H                 ;CHUYEN_BCD CBND1
MOV R1,#7EH
LCALL     CHUYEN_BCD_NH_DO
MOV R0,#1AH                 ;CHUYEN_BCD CBND2
MOV R1,#7AH
LCALL     CHUYEN_BCD_NH_DO
MOV R0,#1CH                 ;CHUYEN_BCD CB3
MOV R1,#76H
LCALL     CHUYEN_BCD_CB_AS
MOV R0,#1EH                 ;CHUYEN_BCD CB4
MOV R1,#73H
```

```
        LCALL       CHUYEN_BCD_CB_AS


THOAT_READING :
        POP   ACC
RET

DOC_AD:
        SETB ALE
        SETB SOC
        CLR ALE
        CLR SOC
TAM3:
        CALL D16NOP
        JNB EOC, TAM3
        CALL D16NOP
DAO_D3_D0:
        MOV  P0,#0FFH
        MOV  A,P0
        MOV  C, ACC.3
        MOV  08H,C
        MOV  C,ACC.0
        MOV  ACC.3,C
        MOV  C, 08H
        MOV  ACC.0, C
 RET

 CONG_DON :
        MOV  B,#5
        DIV   AB
        ADDC        A,@R0                   ;CONG DON PHAN NGUYEN
1
        MOV  @R0,A
        MOV  A,B
        INC   R0
        ADDC        A,@R0                   ;CONG DON PHAN DU   1
        MOV  @R0,A
  RET

  LAM_TRON:
        INC   R0
        MOV  A,@R0                          ;LUU A
        MOV  B,#5
```

```
        DIV   AB
        DEC   R0
        ADDC        A,@R0
        MOV  @R0,A
        MOV  A,B
        CLR   C
        SUBB A,#3                       ;KTRA THAP PHAN > 0.5
        JC     THOAT_LAM_TRON
        INC    @R0
THOAT_LAM_TRON:
RET
                                        ; LAM TRON MOT DON VI
CHUYEN_BCD_NH_DO :
        MOV  A,@R0
        MOV  B, #200
        DIV   AB
        MOV  @R1,A
        MOV  A,B
        MOV  B,#20
        DIV   AB
        DEC   R1
        MOV  @R1,A
        MOV  A,B
        MOV  B,#2
        DIV   AB
        DEC   R1
        MOV  @R1,A
        DEC   R1
        MOV  @R1,B

        MOV  R4,#5

        MOV  @R0,#0       ;KHOI PHUC PHAN NGUYEN,PHAN DU
        INC   R0
        MOV  @R0,#0
  RET
  CHUYEN_BCD_CB_AS :
        MOV  A,@R0
        MOV  B, #100
        DIV   AB
        MOV  @R1,A
        MOV  A,B
        MOV  B,#10
```

```
        DIV   AB
        DEC   R1
        MOV  @R1,A
        DEC   R1
        MOV  @R1,B

        MOV  R4,#5

        MOV  @R0,#0        ;KHOI PHUC PHAN NGUYEN,PHAN DU
        INC   R0
        MOV  @R0,#0
RET

CHUYEN_BCD:
        MOV  A,@R0
        MOV  B, #100
        DIV   AB
        MOV  @R1,A
        MOV  A,B
        MOV  B,#10
        DIV   AB
        DEC   R1
        MOV  @R1,A
        DEC   R1
        MOV  @R1,B
RET
        ;MOV A, R0
        ;MOV C, ACC.0
        ;MOV D0, C
        ;MOV C, ACC.1
        ;MOV D1, C
        ;MOV C, ACC.2
        ;MOV D2, C
        ;MOV C, ACC.3
        ;MOV D3, C


HIENTHI:

XUAT1:
        CLR   LED1
        MOV  A, 7DH
```

```
        MOV  C, ACC.0
        MOV  D0, C
        MOV  C, ACC.1
        MOV  D1, C
        MOV  C, ACC.2
        MOV  D2, C
        MOV  C, ACC.3
        MOV  D3, C
DONG_MO_CUA:

        JNB   CO_DONG_MO_CUA_TU_PC, KTRA_D1
        DJNZ R5,TAM12
        SETB C
        MOV  DONG_MO,C
        MOV  R5,#22
        CLR   CO_DONG_MO_CUA_TU_PC
        MOV  C,CO_DONG_MO_CUA_TU_PC
        MOV  06H,C
        SJMP KTRA_D1
                                ;KHOI TAO LAI THOI GIAN BIT DONG_MO

 TAM12:
        ;CPL   1CH
        ;MOV C,1CH
        MOV  C,CO_DONG_MO_CUA_TU_PC
        MOV  06H,C
        CPL   CO_DONG_MO_CUA_TU_PC
        MOV  C,CO_DONG_MO_CUA_TU_PC
        MOV  DONG_MO,C
        CPL   CO_DONG_MO_CUA_TU_PC



  KTRA_D1:
        JNB   CO_DKHIEN_TU_PC, KTRA_REMOTE_D1
        MOV  C,18H
        MOV  DEN1,C
        MOV  00H,C
  KTRA_REMOTE_D1:
        MOV  C,REMOTE_D1
        JNC   KTRA_CO_DKHIEN_D1
        SETB 7FH
        SJMP KTRA_D2
```

```
KTRA_CO_DKHIEN_D1:
        JNB   7FH,KTRA_D2
        SJMP PHAT_DKHIEN_D1
PHAT_DKHIEN_D1:
        CPL   DEN1
        CPL   00H
        MOV  C,DEN1                          ;LUU TRANG THAI D1
        ;MOV 00H,C
        CLR   7FH

 KTRA_D2:
        JNB   CO_DKHIEN_TU_PC, KTRA_REMOTE_D2
        MOV  C,19H
        MOV  DEN2,C
        MOV  01H,C
 KTRA_REMOTE_D2:
        MOV  C,REMOTE_D2
        JNC   KTRA_CO_DKHIEN_D2
        SETB 7EH
        SJMP  THOAT_KTRA_HALF_REMOTE

 KTRA_CO_DKHIEN_D2:
        JNB   7EH,THOAT_KTRA_HALF_REMOTE
        SJMP PHAT_DKHIEN_D2
 PHAT_DKHIEN_D2:
        CPL   DEN2
        CPL   01H
        ;MOV         C,DEN2
        ;MOV 01H,C
        CLR   7EH

  THOAT_KTRA_HALF_REMOTE:
        LCALL DELAY
        SETB LED1
  XUAT2 :
        CLR   LED2
        MOV  A, 7CH
        MOV  C, ACC.0
        MOV  D0, C
        MOV  C, ACC.1
        MOV  D1, C
        MOV  C, ACC.2
```

```
        MOV D2, C
        MOV C, ACC.3
        MOV D3, C

KTRA_CHUONG:
        MOV C,CHUONG
        MOV 05H,C
KTRA_TR_THAI_CUA:
        MOV C,TR_THAI_CUA
        MOV 04H,C
KTRA_Q1:
        JNB  CO_DKHIEN_TU_PC, KTRA_REMOTE_Q1
        MOV C,1AH
        MOV QUAT1,C
        MOV 02H,C
KTRA_REMOTE_Q1:
        MOV C,REMOTE_Q1
        JNC  KTRA_CO_DKHIEN_Q1
        SETB 7DH
        SJMP KTRA_Q2

KTRA_CO_DKHIEN_Q1:
        JNB  7DH,KTRA_Q2
        SJMP PHAT_DKHIEN_Q1
PHAT_DKHIEN_Q1:
        CPL  QUAT1
        CPL  02H
        ;MOV        C,QUAT1
        ;MOV 02H,C
        CLR  7DH

KTRA_Q2:
        JNB  CO_DKHIEN_TU_PC, KTRA_REMOTE_Q2
        MOV C,1BH
        MOV QUAT2,C
        MOV 03H,C
        CLR  CO_DKHIEN_TU_PC
KTRA_REMOTE_Q2:
        MOV C,REMOTE_Q2
        JNC  KTRA_CO_DKHIEN_Q2
        SETB 7CH
        SJMP THOAT_KTRA_REMOTE
```

```
KTRA_CO_DKHIEN_Q2:
        JNB   7CH,THOAT_KTRA_REMOTE
        SJMP PHAT_DKHIEN_Q2
PHAT_DKHIEN_Q2:
        CPL   QUAT2
        CPL   03H
        ;MOV          C,QUAT2
        ;MOV 03H,C
        CLR   7CH
 THOAT_KTRA_REMOTE:
        LCALL DELAY
        SETB LED2
        LJMP XUAT1
 RET

 LUU_TTHAI_THBI:
 PUSH ACC
 MOV  R0,#20H
 MOV  R1,#70H
 LCALL CHUYEN_BCD
 POP   ACC
 RET

 NHAN_DKHIEN_TU_PC:
 PUSH ACC
 MOV  R0,#23H
 ;MOV R1,#6EH
 ;LCALL CHUYEN_BCD
 POP   ACC
 RET

  WAIT :

  MOV  TH0, #80H
  MOV  TL0, #01H
  SETB TR0
  WAIT1:
  JNB TF0, WAIT1
  CLR TF0
  CLR TR0
  DJNZ R3, WAIT
  MOV R3, #17
  RET
```

```
DELAY:
        MOV  R7,#200
        MOV  R6,#10

LOOP:
        DJNZ R7,$
        MOV  R7,#200
        DJNZ R6,LOOP

 RET

 D16NOP :
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
         NOP
  RET
  TIMER_HALF_S :
          DJNZ R3, THOAT_TIMER
          CLR TR0
          LCALL READING
          LCALL LUU_TTHAI_THBI
          LCALL       NHAN_DKHIEN_TU_PC
          MOV  R0, #7FH
          SETB TI
          SETB TR0
          MOV R3, #17
   THOAT_TIMER:
   RETI
```

```
SERIAL:
PUSH ACC
MOV DPTR,#ASCII
KTRA_TI:

JNB   TI, KTRA_RI
CJNE R0, #6CH, PHAT
SJMP  KTRA_RI
PHAT:
MOV  A, @R0
MOVC A,@A+DPTR
MOV  SBUF, A
DEC   R0
CLR   TI

KTRA_RI:
JNB   RI, THOAT_SERIAL
MOV  23H,SBUF
SETB CO_DKHIEN_TU_PC
CLR   RI
THOAT_SERIAL :
POP   ACC
CLR   TI
RETI


ASCII : DB '0123456789ABCDEF'

END
```

Mã nguồn của phần mềm trên máy tính :

```
Dim am_baodong As String, am_chuong As String
Dim so_lan_phat_chuong As Integer
Dim co_chophep_baodong1 As Boolean, co_chophep_baodong2 As Boolean,
co_chophep_dkh_thbi As Boolean
Dim co_chophep_tddkh_den1 As Boolean, co_chophep_tddkh_den2 As Boolean,
co_chophep_tddkh_quat1 As Boolean, co_chophep_tddkh_quat2 As Boolean
Dim chophep_phat_baodong1 As Boolean, chophep_phat_baodong2 As Boolean
Dim STX As Boolean, ETX As Boolean
Dim SO_KY_TU_THU As String
Dim STR As String, STR2 As String, D1 As String, D2 As String, Q1 As String, Q2 As
String, TR_THAI_CHUONG As String, TR_THAI_CUA As String, DKTB As String

 Public Function BinToDec(BinNum As String) As String
   Dim i As Integer
   Dim DecNum As Long
    For i = Len(BinNum) To 1 Step -1
      If Mid(BinNum, i, 1) And 1 Then
       DecNum = DecNum + 2 ^ (Len(BinNum) - i)
      End If
   Next i

   BinToDec = DecNum

 End Function

 Public Function DecToBin(DecNum As String) As String
   Dim BinNum As String
   Dim lDecNum As Long
   Dim i As Integer

   i = 0
   lDecNum = Val(DecNum)

   Do
     If lDecNum And 2 ^ i Then
       BinNum = "1" & BinNum
     Else
       BinNum = "0" & BinNum
     End If
     i = i + 1
   Loop Until i = 8
```

```
' Return BinNum as a String
  DecToBin = BinNum

End Function


Private Sub as1_Change()
If co_chophep_dkh_thbi Then
   If Val(as1.Text) > 250 Then
      BAT_D1_Click
   End If
   If Val(as1.Text) < 2 Then
      TAT_D1_Click
   End If

End If
End Sub

Private Sub as2_Change()

If co_chophep_dkh_thbi Then
   If Val(as2.Text) > 11 Then
      BAT_D2_Click
   End If
   If Val(as2.Text) < 8 Then
      TAT_D2_Click
   End If


End If

End Sub

Private Sub BAT_D1_Click()
DKTB = TR_THAI_CHUONG & TR_THAI_CUA & Q2 & Q1 & D2 & "1"
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub BAT_D2_Click()
DKTB = Q2 & Q1 & "1" & D1
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
```

```
End Sub

Private Sub BAT_HET_Click()
DKTB = "1111"
dkc.Output = Chr(BinToDec(DKTB))

End Sub

Private Sub BAT_Q1_Click()
DKTB = Q2 & "1" & D2 & D1
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub BAT_Q2_Click()
DKTB = "1" & Q1 & D2 & D1
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub chophep_baodong_Click()


If chophep_phat_baodong1 Or chophep_phat_baodong2 Then
   chophep_baodong.Enabled = False
Else
   co_chophep_baodong1 = Not co_chophep_baodong1
   co_chophep_baodong2 = Not co_chophep_baodong2
   co_chophep_dkh_thbi = Not co_chophep_dkh_thbi
    If co_chophep_dkh_thbi = True Then
      co_chophep_tddkh_den1 = True
      co_chophep_tddkh_den2 = True
      co_chophep_tddkh_quat1 = True
      co_chophep_tddkh_quat2 = True
    Else
      co_chophep_tddkh_den1 = False
      co_chophep_tddkh_den2 = False
      co_chophep_tddkh_quat1 = False
      co_chophep_tddkh_quat2 = False
    End If

   If co_chophep_baodong1 And co_chophep_baodong2 And co_chophep_dkh_thbi
Then
   MsgBox "CHE DO TU DONG DANG HOAT DONG"
   trangthai_hethong.Caption = "CHE DO AUTO"
```

```
  Else
  MsgBox "BAN DA TAT CHE DO TU DONG "
  trangthai_hethong.Caption = "CHE DO MANUAL"
  End If
End If
End Sub

Private Sub CHUONG_Change()
If CHUONG.Text = "0" Then
   HINHCHUONG.Visible = True
   so_lan_phat_chuong = 2
   PHATCHUONG.URL = am_chuong
   PHATCHUONG.Controls.play
End If
End Sub

Private Sub data_Change()
Dim tthai_thbi As String
If (Mid(data.Text, 4, 1) = 1) Then

temp1.Text = Mid(data.Text, 2, 2) & ".5"
Else

temp1.Text = Mid(data.Text, 2, 2)
End If

If (Mid(data.Text, 8, 1) = 1) Then

temp2.Text = Mid(data.Text, 6, 2) & ".5"
Else

temp2.Text = Mid(data.Text, 6, 2)
End If


as1.Text = Mid(data.Text, 9, 3)
as2.Text = Mid(data.Text, 12, 3)

tthai_thbi = Mid(data.Text, 15, 3)

D1 = Mid(DecToBin(tthai_thbi), 8, 1)
D2 = Mid(DecToBin(tthai_thbi), 7, 1)
Q1 = Mid(DecToBin(tthai_thbi), 6, 1)
```

```
Q2 = Mid(DecToBin(tthai_thbi), 5, 1)
TR_THAI_CUA = Mid(DecToBin(tthai_thbi), 4, 1)
TR_THAI_CHUONG = Mid(DecToBin(tthai_thbi), 3, 1)

If TR_THAI_CUA = "1" Then
   CUA.Text = "CHUA DONG"
   Else
   CUA.Text = "DA DONG"
End If
CHUONG.Text = TR_THAI_CHUONG
If D1 = "1" Then
DEN1.Text = "MO"
 Else
 DEN1.Text = "TAT"
 End If
If D2 = "1" Then
DEN2.Text = "MO"
 Else
 DEN2.Text = "TAT"
 End If

 If Q1 = "1" Then
 QUAT1.Text = "MO"
  Else
  QUAT1.Text = "TAT"
  End If

 If Q2 = "1" Then
 QUAT2.Text = "MO"
  Else
  QUAT2.Text = "TAT"
  End If
 End Sub


  Private Sub dkc_OnComm()

  If (dkc.CommEvent = 2) Then

    STR2 = dkc.Input

      If STR2 = "A" Then
```

```
    STX = True
  End If
  If STX = True And STR2 = "B" Then

      ETX = True
  End If
  If STX = True Then
  STR = STR & STR2
  End If
  If ETX = True Then
     data.Text = Mid(STR, 2, Len(STR) - 2)
      STR = ""
      STX = False
     ETX = False
    End If
End If

End Sub
```

```
Private Sub DONG_MO_CUA_Click()

DKTB = "1" & Q2 & Q1 & D2 & D1
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
'MsgBox BinToDec(DKTB)
End Sub

Private Sub Form_KeyPress(keyascii As Integer)
dkc.Output = Chr(keyascii)


End Sub

Private Sub Form_Load()

SO_KY_TU_THU = 17

STX = False
ETX = False
am_chuong = "E:\my document\download\my\dl_khongtangtoc\doorbell2.wav"
am_baodong = "E:\my document\download\my\dl_khongtangtoc\alarm.wav"
co_chophep_baodong1 = True
co_chophep_baodong2 = True
co_chophep_dkh_thbi = True
chophep_phat_baodong1 = False
chophep_phat_baodong2 = False
dkc.CommPort = 1
dkc.OutBufferSize = 2098
dkc.Settings = "9600,n,8,1"
dkc.Handshaking = comNone
dkc.InBufferSize = 2000
dkc.PortOpen = True
dkc.InputLen = 1
dkc.DTREnable = True
dkc.EOFEnable = False
dkc.NullDiscard = True
dkc.InputMode = comInputModeText
dkc.RThreshold = 1
dkc.RTSEnable = True
dkc.SThreshold = 8
End Sub
```

```
Private Sub PHATCHUONG_PlayStateChange(ByVal NewState As Long)
Dim so_lan_phat As Integer
If NewState = 8 And so_lan_phat_chuong <> 0 Then
   Timer_PHATCHUONG.Enabled = True
   so_lan_phat_chuong = so_lan_phat_chuong - 1


If so_lan_phat_chuong = 0 Then
   HINHCHUONG.Visible = False
End If
End If

End Sub

Private Sub Tat_am_baodong_Click()

chophep_phat_baodong1 = False
chophep_phat_baodong2 = False
WindowsMediaPlayer1.Controls.stop
WindowsMediaPlayer2.Controls.stop

Timer1.Enabled = False
Timer2.Enabled = False
co_chophep_baodong1 = True
co_chophep_baodong2 = True
 chophep_baodong.Enabled = True
End Sub

Private Sub TAT_D1_Click()
DKTB = Q2 & Q1 & D2 & "0"
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub TAT_D2_Click()
DKTB = Q2 & Q1 & "0" & D1
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub TAT_HET_Click()
DKTB = "0000"
```

```
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub TAT_Q1_Click()
DKTB = Q2 & "0" & D2 & D1
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub TAT_Q2_Click()
DKTB = "0" & Q1 & D2 & D1
'DKTB = "1101"
dkc.Output = Chr(BinToDec(DKTB))
End Sub

Private Sub temp1_Change()
If co_chophep_dkh_thbi Then
   If Val(temp1.Text) > 33 Then
      BAT_Q1_Click
   End If
   If Val(temp1.Text) < 32 And Not chophep_phat_baodong1 Then
      TAT_Q1_Click
   End If
   If co_chophep_baodong1 And Val(temp1.Text) > 34 Then
      WindowsMediaPlayer1.URL = am_baodong
      WindowsMediaPlayer1.Controls.play
      co_chophep_baodong1 = False
      chophep_phat_baodong1 = True
   End If
End If
End Sub

Private Sub temp2_Change()
If co_chophep_dkh_thbi Then
   If Val(temp2.Text) > 33 Then
      BAT_Q2_Click
   End If
   If Val(temp2.Text) < 32 And Not chophep_phat_baodong2 Then
      TAT_Q2_Click
   End If
   If co_chophep_baodong2 And Val(temp2.Text) > 34 Then
      WindowsMediaPlayer2.URL = am_baodong
      WindowsMediaPlayer2.Controls.play
```

```
        co_chophep_baodong2 = False
        chophep_phat_baodong2 = True
    End If
End If
End Sub

Private Sub Timer_PHATCHUONG_Timer()
PHATCHUONG.Controls.play
Timer_PHATCHUONG = False
End Sub

Private Sub Timer1_Timer()
WindowsMediaPlayer1.Controls.play
Timer1.Enabled = False
End Sub

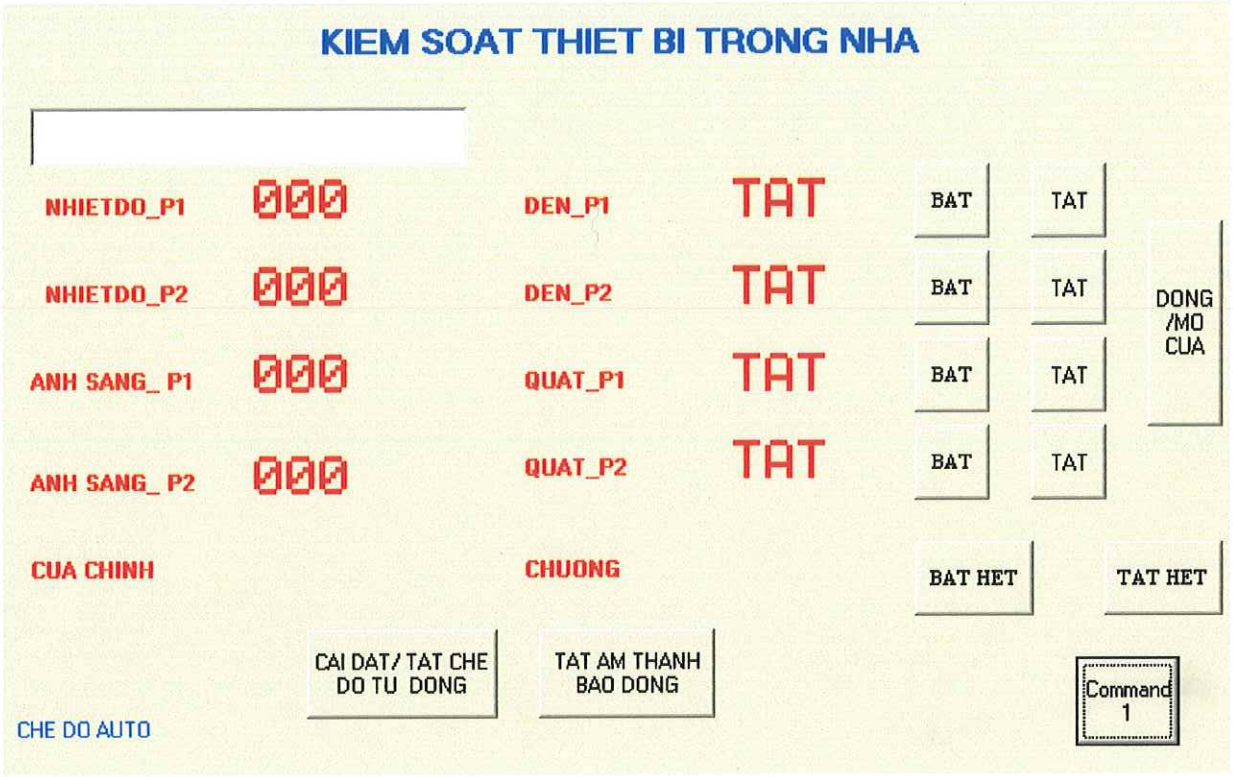
Private Sub Timer2_Timer()
WindowsMediaPlayer2.Controls.play
Timer2.Enabled = False
End Sub

Private Sub WindowsMediaPlayer1_PlayStateChange(ByVal NewState As Long)
If NewState = 8 And chophep_phat_baodong1 Then
Timer1.Enabled = True
End If
End Sub

Private Sub WindowsMediaPlayer2_PlayStateChange(ByVal NewState As Long)
If NewState = 8 And chophep_phat_baodong2 Then
Timer2.Enabled = True
End If
End Sub


Private Sub WindowsMediaPlayer3_OpenStateChange(ByVal NewState As Long)
If NewState = 8 And chophep_phat_baodong2 Then
Timer2.Enabled = True
End If
End Subrack
```

Giao diện phần mềm trên máy tính :



TÀI LIỆU THAM KHẢO

Tài liệu tham khảo

Tiếng Việt

[1] Lê Đăng Đức, Hà Đắc Bình (2004), Kỹ thuật vi xử lí, Đại học GTVT TP.HCM.

[2] Nguyễn Văn Hùng (2007), Hệ thống tự động điều khiển trong ngôi nhà thông minh, Đại học kỹ thuật công nghệ TP.HCM.

[3] Lưu Phú- Lê Phi Yến., Kỹ thuật điện tử (1995), Đại học Bách Khoa TP.HCM.

[4] Phạm Hồng Sơn, Trần Hoài An, Kỹ thuật số (2003), Đại học GTVT TP.HCM.

[5] Ngô Diên Tập (2001), Lập trình ghép nối máy tính trong Windows, Nxb Khoa học kỹ thuật.

Tiếng Anh

[6] Jan Axelson (1998), Serial Port Complete, Lakeview Research.

[7] Paul Scherz (2000), Practical Electronics for Inventors, McGraw-Hill.

Trang web tham khảo

en.wikipedia.org

smarthome.com