

HƯỚNG DẪN SỬ DỤNG PHẦN MỀM MÔ PHỎNG PROTEUS GIỚI THIỆU PHẦN MỀM

Proteus là một phần mềm hỗ trợ thiết kế và mô phỏng các loại mạch điện tử. So với một số các phần mềm hỗ trợ mô phỏng các mạch điện khác như:

- *Circuitmaker 2000* ngoài việc hỗ trợ thiết kế sơ đồ mạch in (PCB) thì phần mềm còn cho phép vẽ và mô phỏng mạch số chuẩn xác tới 95% nhưng lại bị hạn chế khi mô phỏng các loại mạch tương tự cũng như mạch tổng hợp các số cả tương tự.

- *WorkBench (EWB)* phần mềm này có thư viện linh kiện phong phú và có nhiều thiết bị đo kiểm tra như thực tế giúp người thiết kế dễ dàng quan sát cân chỉnh thông số của mạch điện. So với các phần mềm khác thì WorkBench cho phép mô phỏng các loại mạch điện (cả số và tương tự)

- ORCAD, Eagle và PROTEL là các phần mềm hỗ trợ vẽ sơ đồ nguyên lý và sơ đồ mạch in (PCB) nhưng không hỗ trợ mô phỏng mạch nguyên lý.

Thì Proteus có thể mạch hơn hẳn về các mặt:

- Thư viện linh kiện phong phú .
- Hỗ trợ nhiều thiết bị đo kiểm tra.
- Cho phép thiết kế và chạy mô phỏng sơ đồ nguyên lý gồm các mạch tương tự, mạch số, mạch tổng hợp cả số cả tương tự...
- Cho phép chạy mô phỏng các loại vi điều khiển, EPPROM, PIC.
- Hỗ trợ thiết kế mạch in (PCB).

Vì các điểm mạch của phần mềm nên đối với giáo viên có thể sử dụng Proteus để làm công cụ hỗ trợ cho các môn học như: Điện tử cơ bản, Lý thuyết mạch, Mạch điện 1, Mạch điện 2, kỹ thuật xung số, môn học Lập trình Vi điều khiển... đối với học sinh – sinh viên đây là công cụ đắc lực phục vụ quá trình học tập, đối với những người yêu thích điện - điện tử thì đây là môi trường sát với thực tế nhất để thiết kế các mạch điện ứng dụng.

Là người giáo viên, là một thợ điện tử tôi thấy phần mềm này rất hay và cần thiết cho nên tôi viết tài liệu này cho bạn đọc ..

Do đây là phiên bản đầu tiên và cũng do thời gian ngắn nên tài liệu chỉ dừng lại ở mức độ hướng dẫn cơ bản giúp những ai mới bắt đầu tìm hiểu về phần mềm này có thể làm ngay được. Còn chi tiết hơn xin ra mắt bạn đọc trong lần tái bản sau. Tài liệu chắc cần nhiều sự góp ý của bạn đọc.

Mọi thông tin phản hồi xin gửi về địa chỉ: dlong_nute_edu_vn@yahoo.com.vn

CHƯƠNG 1: HƯỚNG DẪN CÀI ĐẶT PHẦN MỀM

Trước khi bắt tay vào sử dụng phần mềm thì cũng giống như các phần mềm khác ta phải tiến hành cài đặt chương trình phần mềm vào máy tính đã. Các bạn có thể tìm mua phần mềm này ở các cửa hàng cung cấp phần mềm điện tử hoặc có thể tìm kiếm trên Internet (<http://www.sonrivi.com>) rồi download về máy để chuẩn bị cài đặt. Tùy thuộc vào điều kiện khách quan của mình mà các bạn lựa chọn các phiên bản cho thích hợp: Hiện nay nhà cung cấp phần mềm này đã đưa ra khá nhiều các phiên bản với nhiều tính năng mới được cải thiện: Hai phiên bản hiện nay đang được sử dụng nhiều là PROTEUS 6.7 Pro SP3 Final và Crack Prosys_6.9.03 . Tuy nhiên phiên bản PROTEUS 6.7 có nhược điểm là chỉ cho phép người thiết kế thực hiện mọi thao tác trong khoảng 15 phút đầu sau đó chương trình mất đi khả năng lưu trữ nội dung thiết kế, để khắc phục nhược điểm đó các bạn có thể sử dụng phiên bản PROTEUS 6.9.03 cho phép thiết kế, mô phỏng rất tốt và thời gian lưu trữ File là không hạn chế. Chính vì các lí do đó mà trong tài liệu này tác giả đã chọn và sử dụng phiên bản PROTEUS 6.9.03 để trình bày với các bạn.

1.1. Yêu cầu cấu hình

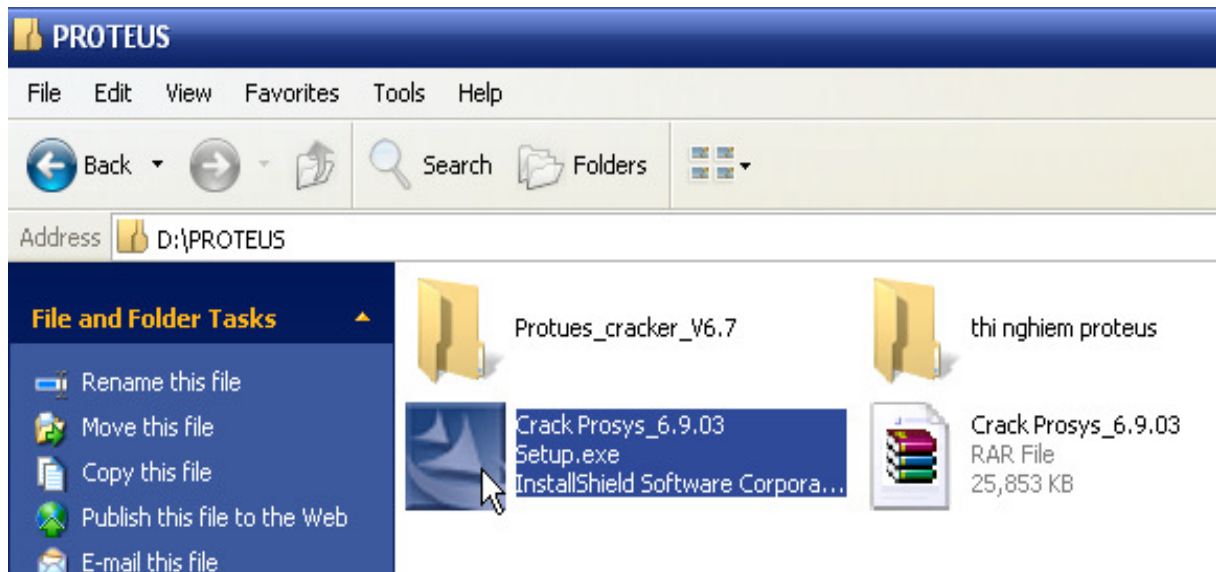
Phần mềm PROTEUS hỗ trợ mạnh việc thiết kế sơ đồ nguyên lý, chạy mô phỏng mạch điện cũng như việc thiết kế mạch in chính vì thế để làm việc được với phần mềm này thì yêu cầu máy tính của bạn phải có cấu hình tối thiểu như sau:

- Bộ xử lý Pentium 1.6 GHz trở lên, Bộ nhớ Ram tối thiểu là 128 Mb, ổ cứng 40 Gb trở lên, ổ CD-ROM 52X, Card AGP 32Mb, màn hình 17 inches trở lên.
- Chạy trên môi trường Windows 2000, NT, hoặc Windows XP.

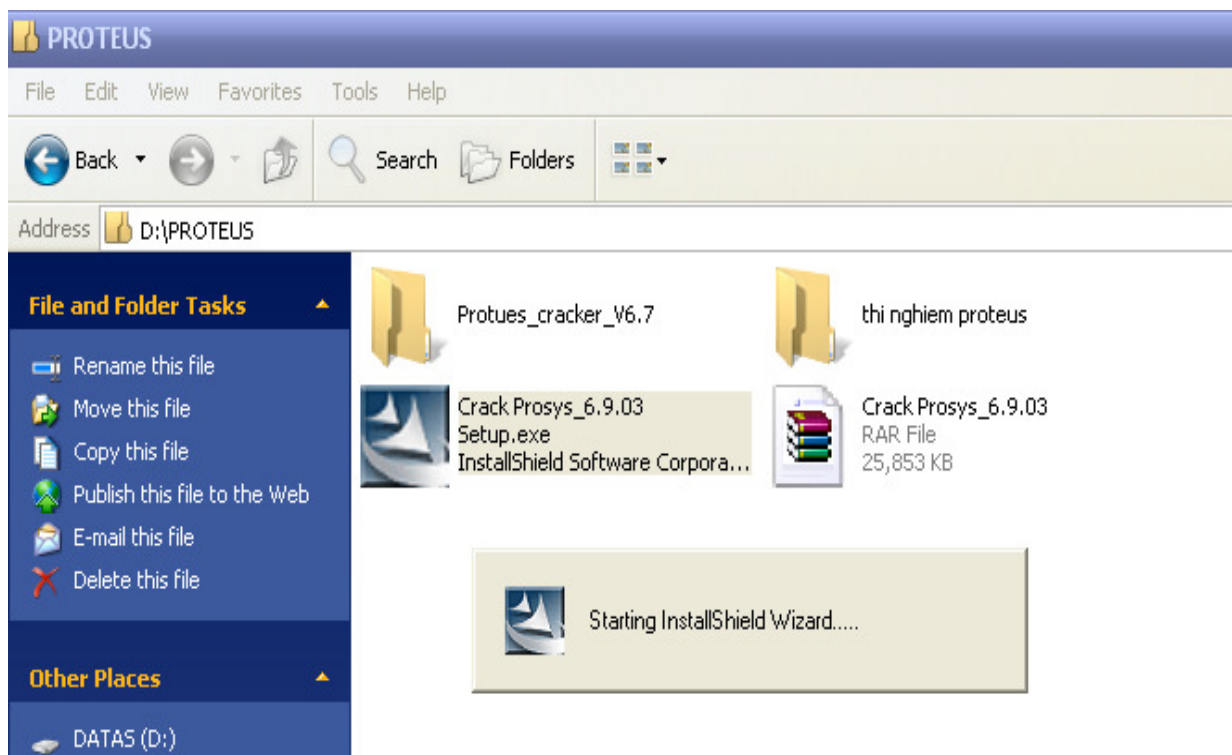
1.2. Các bước cài đặt phần mềm lên đĩa cứng

Việc cài đặt có thực hiện trực tiếp trên đĩa CD hoặc trên đĩa cứng ở đây tác giả hướng dẫn cách cài đặt phần mềm được lưu trữ sẵn trên đĩa cứng của máy (trên ổ D\ PROTEUS) theo thứ tự như sau:

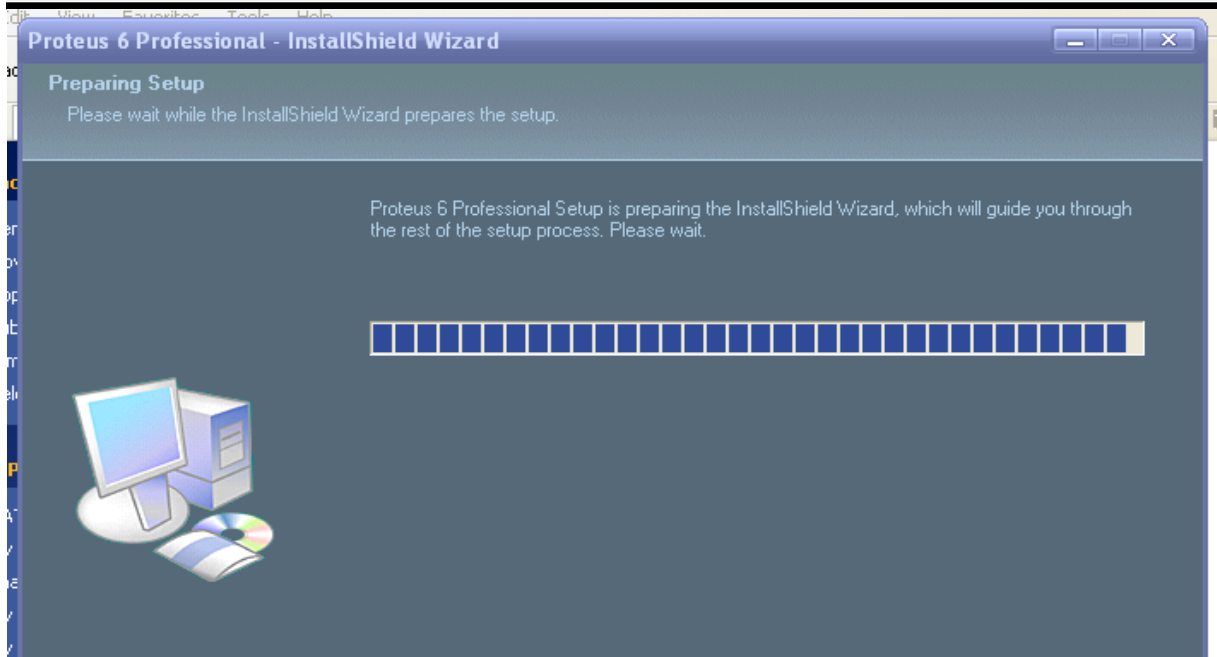
- Vào ổ D theo đường dẫn: D\PROTEUS sau đó nhấp đúp trái chuột vào thư mục *Crack Prosys_6.9.03* như sau:



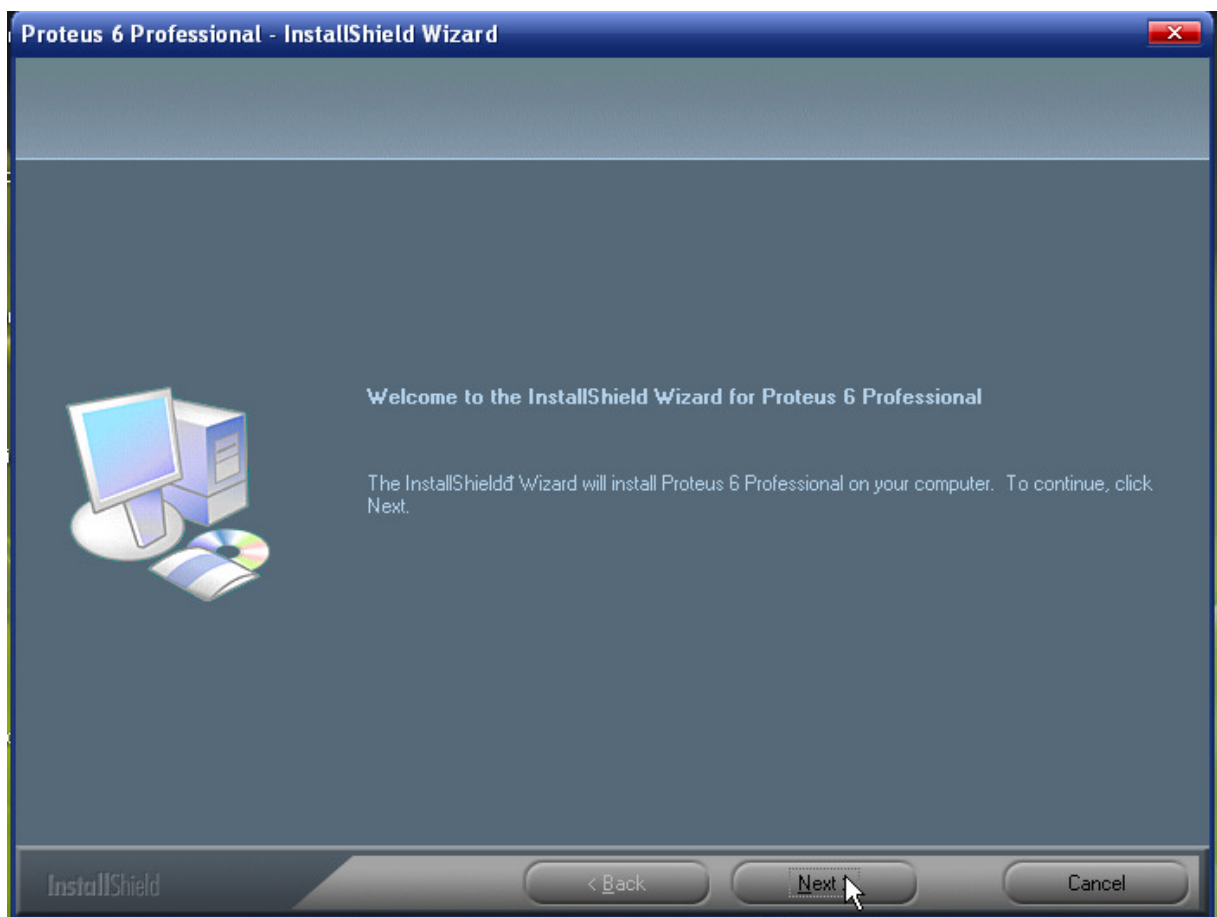
Khi đó một cửa sổ mới xuất hiện với tiêu đề **Starting InstallShield Wizard** thông báo rằng quá trình cài đặt bắt đầu



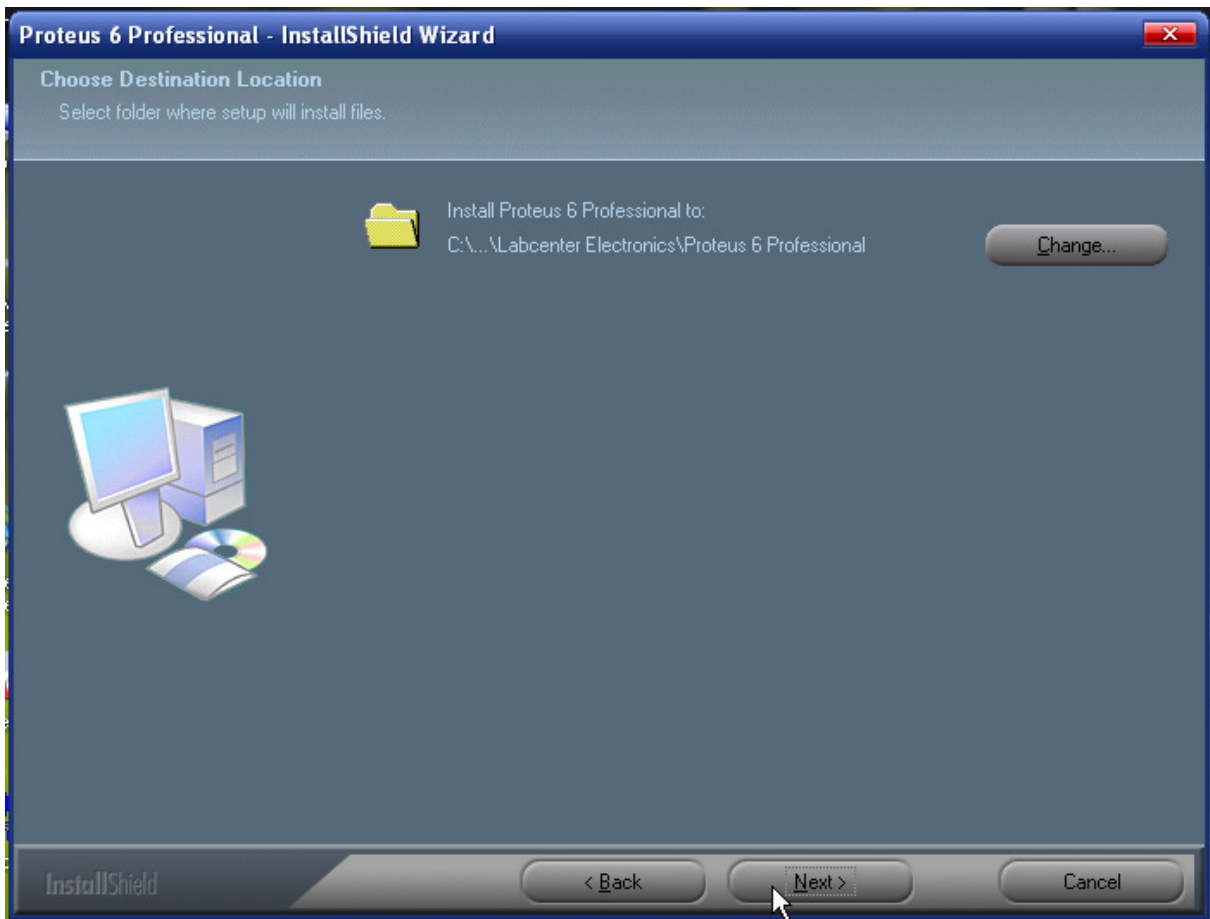
Sau quá trình này một cửa sổ tiếp theo xuất hiện thông báo cho bạn biết việc cài đặt đang được chuẩn bị (Preparing Setup) và yêu cầu bạn chờ đợi trong giây lát (Please wait while the Starting InstallShield Wizard prepares the setup).



Sau khi quá trình chuẩn bị cài đặt đã thành công một cửa sổ tiếp xuất hiện với chỉ dẫn: The InstallShield Wizard will Install Proteus 6 Professional on your computer. To continue, click Next - bạn hãy nháy chuột lên buton **Next** để tiếp tục quá trình cài đặt Proteus 6 Professional lên máy tính.

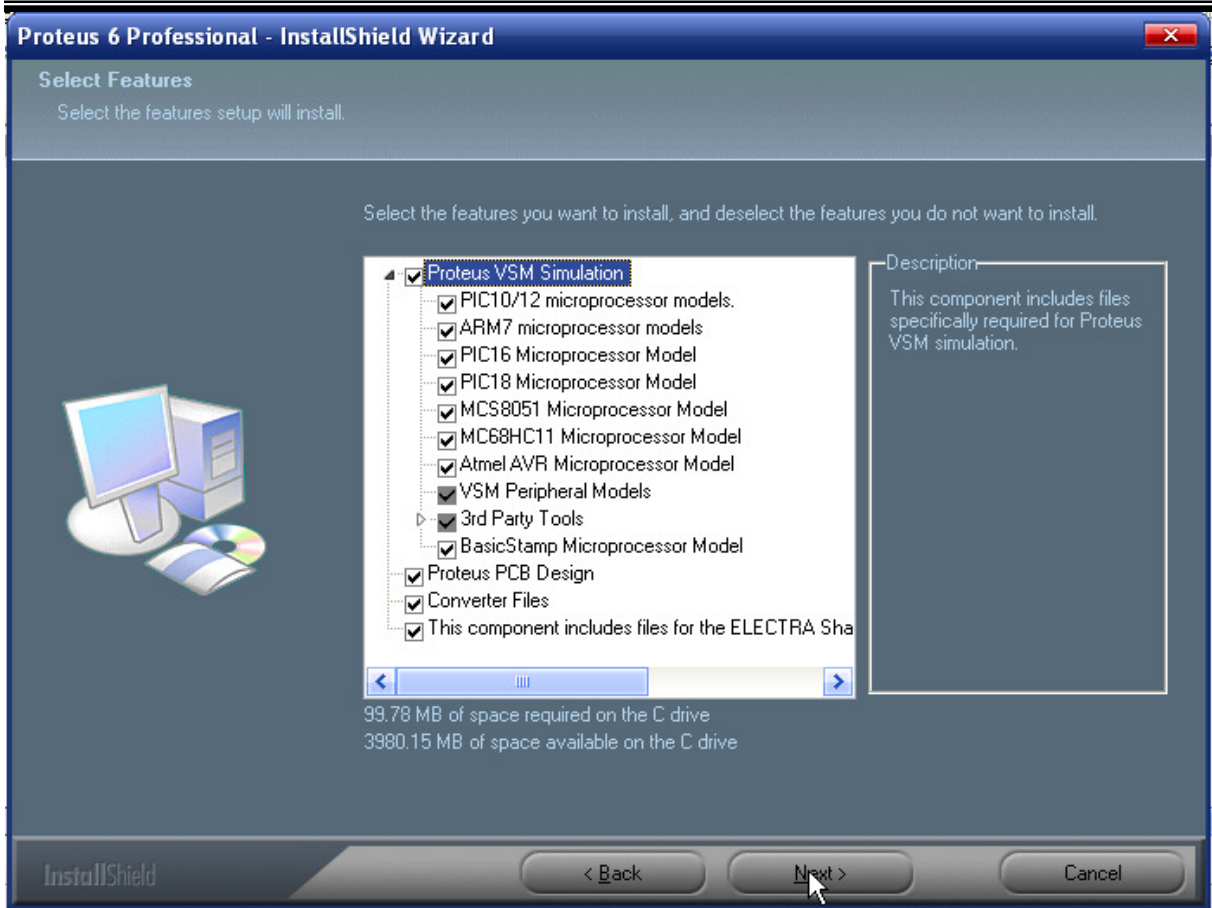


Cửa sổ tiếp theo xuất hiện thông báo cho bạn biết vị trí lưu trữ thư mục cài đặt – theo mặc định của chương trình thì thư mục cài đặt sẽ được lưu trong ổ C \ Program Files (tuy nhiên bạn có thể thay đổi vị trí lưu trữ này bằng cách nhấp chuột vào button **Change** và làm theo chỉ dẫn) sau đó nhấp chuột vào button **Next** để tiếp tục.

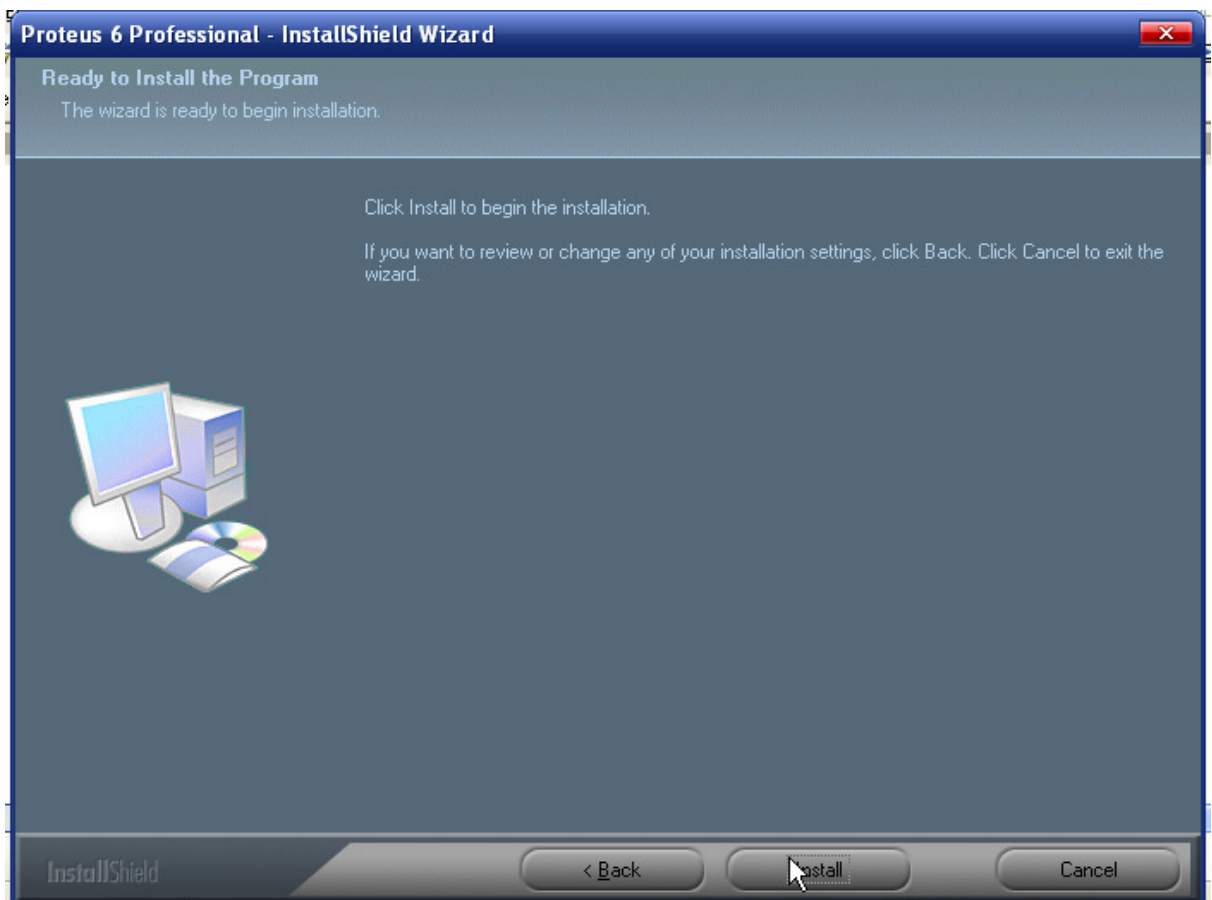


Sau khi nhấp chọn Next một cửa sổ mới tiếp theo hiện ra trên cửa sổ này có chỉ dẫn rằng *Select the Features setup will install* – lựa chọn các chỉ mục được phép cài đặt. Theo mặc định của chương trình thì tất cả các chỉ mục đều được chọn để cài đặt mô phỏng với tổng dung lượng là 99.78 MB và để giải phóng cho ổ cứng của PC bạn có thể deselect (không chọn) những chỉ mục mà mình ít hoặc chưa có thời gian thao tác tới nó. Nhưng theo ý tác giả thì với cấu hình máy mà ta đã lựa chọn như ban đầu thì 99.78MB/40GB là không đáng kể gì vì thế chúng ta nên chọn tất cả theo mặc định.

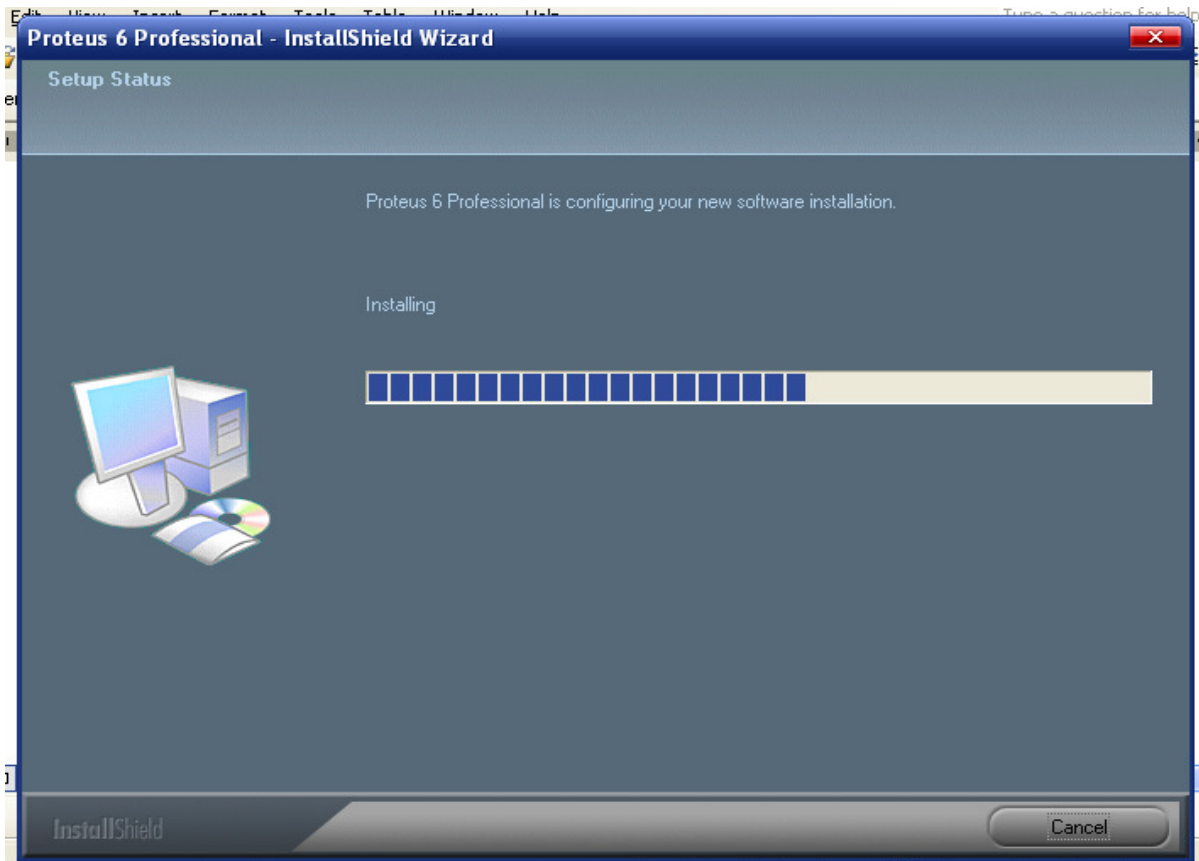
Khi bạn đã thiết lập các lựa chọn xong để tiếp tục quá trình cài đặt bạn tiếp tục nhấp chuột chọn **Next**.



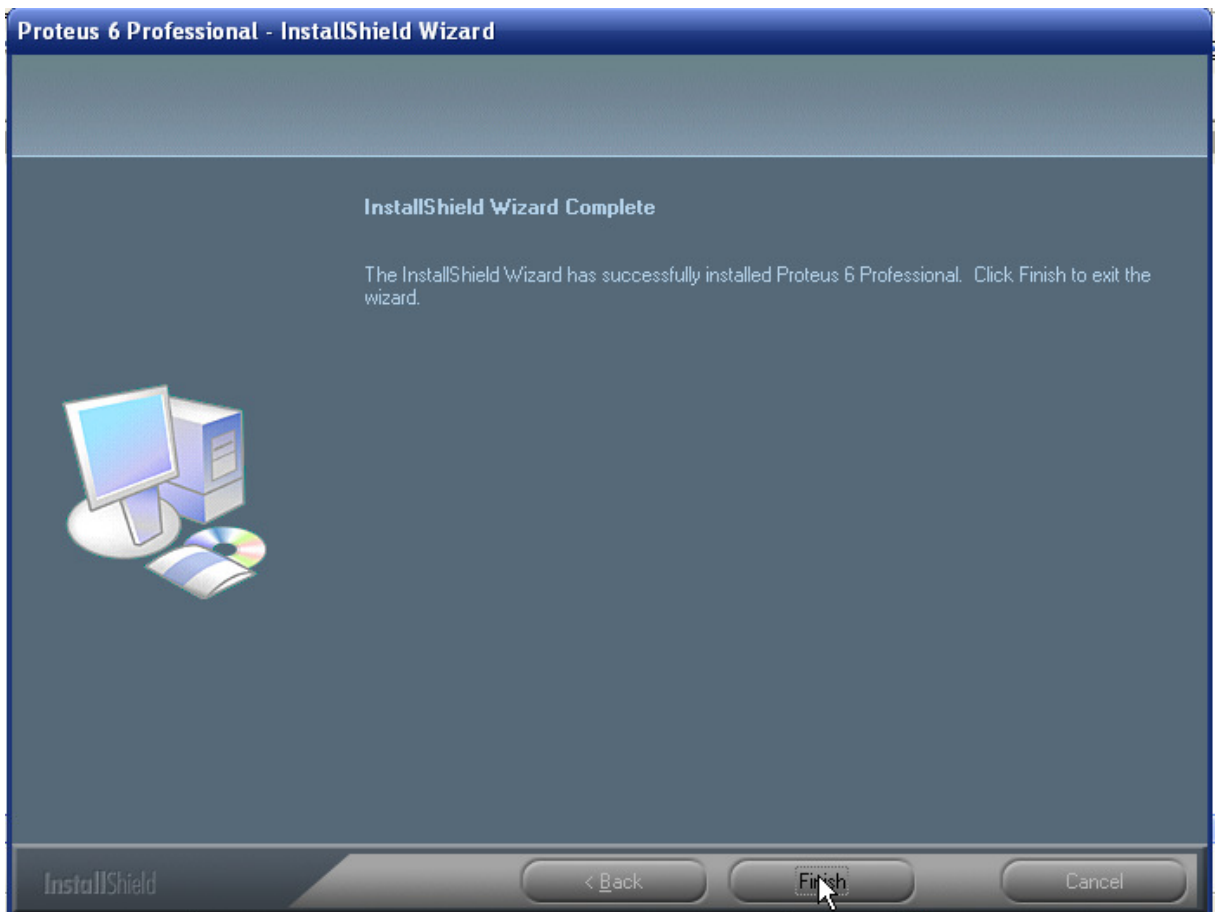
Tiếp theo nháy chọn button **Install** để bắt đầu quá trình cài đặt.



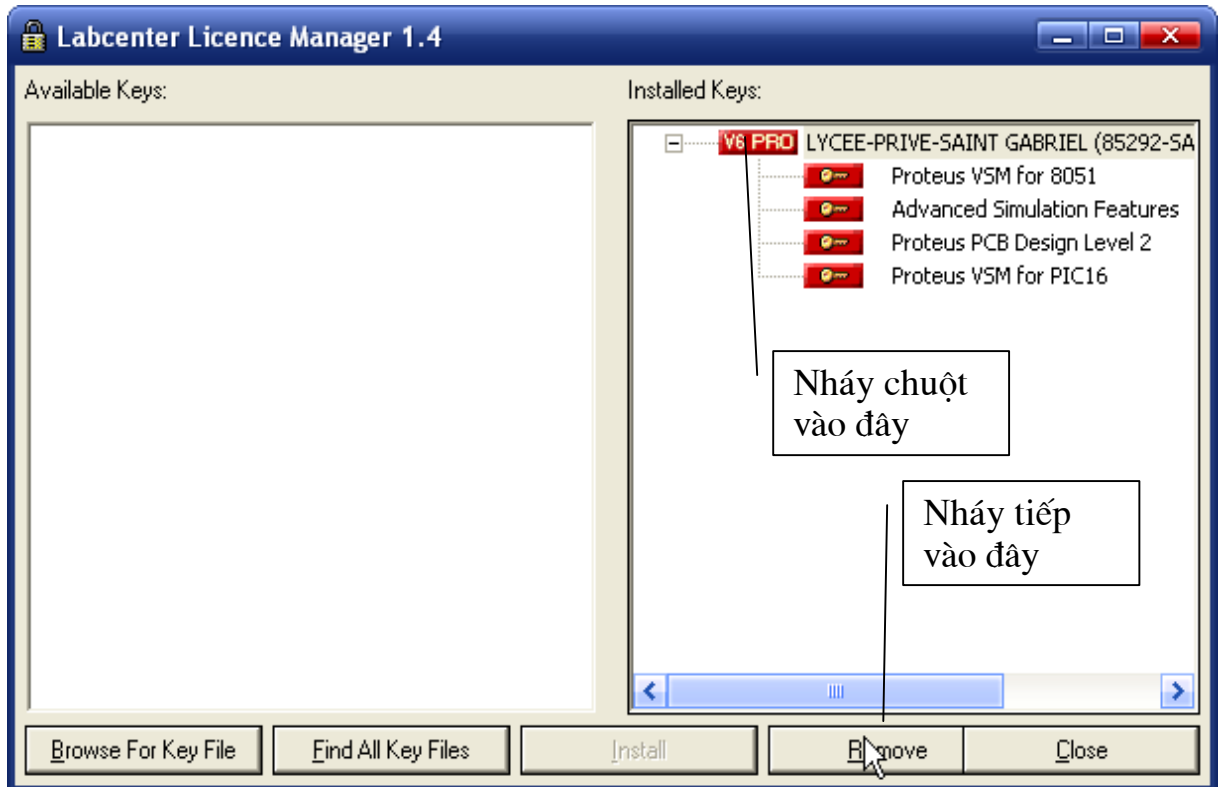
Quá trình cài đặt đang thực hiện bạn hãy đợi giây lát



Sau khi quá trình cài đặt thành công nháy chọn button **Finish** để kết thúc.



- Nếu sau khi cài đặt xong mà bạn không thể khởi động để vào chương trình được thì bạn hãy làm như sau để **Crack**:
- Bạn vào ổ đĩa C\ Program Files \ Labcenter Electronics \ Proteus 6 Professional \ BIN sau đó nháy đúp chuột vào LICENCE.EXE một cửa sổ hiện ra như sau:



Bạn làm tiếp như sau: trong ô Installed Keys chọn dòng trên cùng --> nháy chuột vào nút Remove. Sau đó trở lại thư mục thực hiện cài đặt bổ sung --> thế là xong. **Chúc bạn thành công!**

1.3. Khởi động và thoát khỏi chương trình

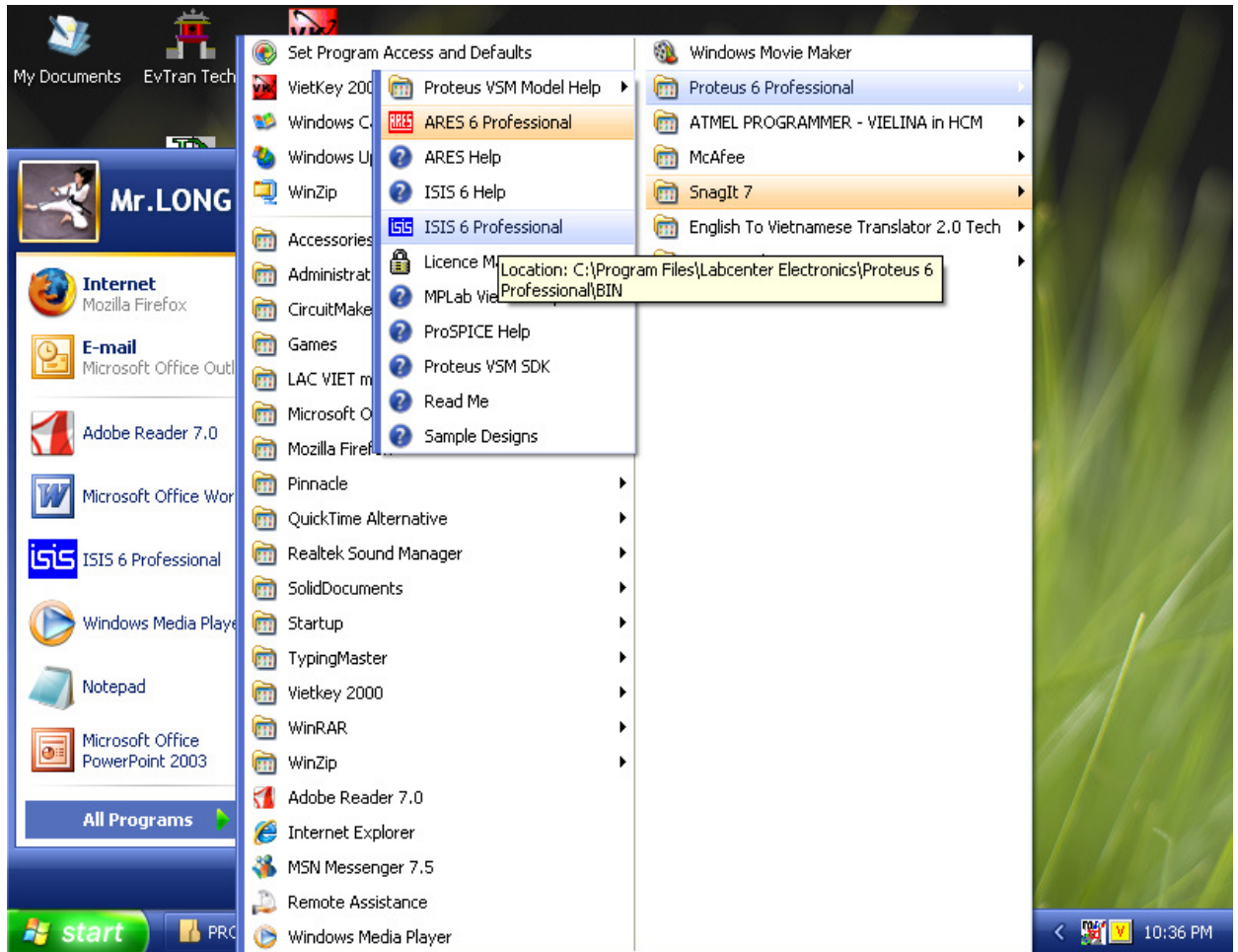
a. Khởi động chương trình

Cũng giống như các phần mềm khác để bắt tay vào công việc thiết kế các bạn phải biết cách khởi động chương trình. Có nhiều cách để mở chương trình Proteus và 2 cách cơ bản nhất là:

- Chọn *Start \ All Programs \ Proteus 6 Profesional \ ISIS 6 Professional*.



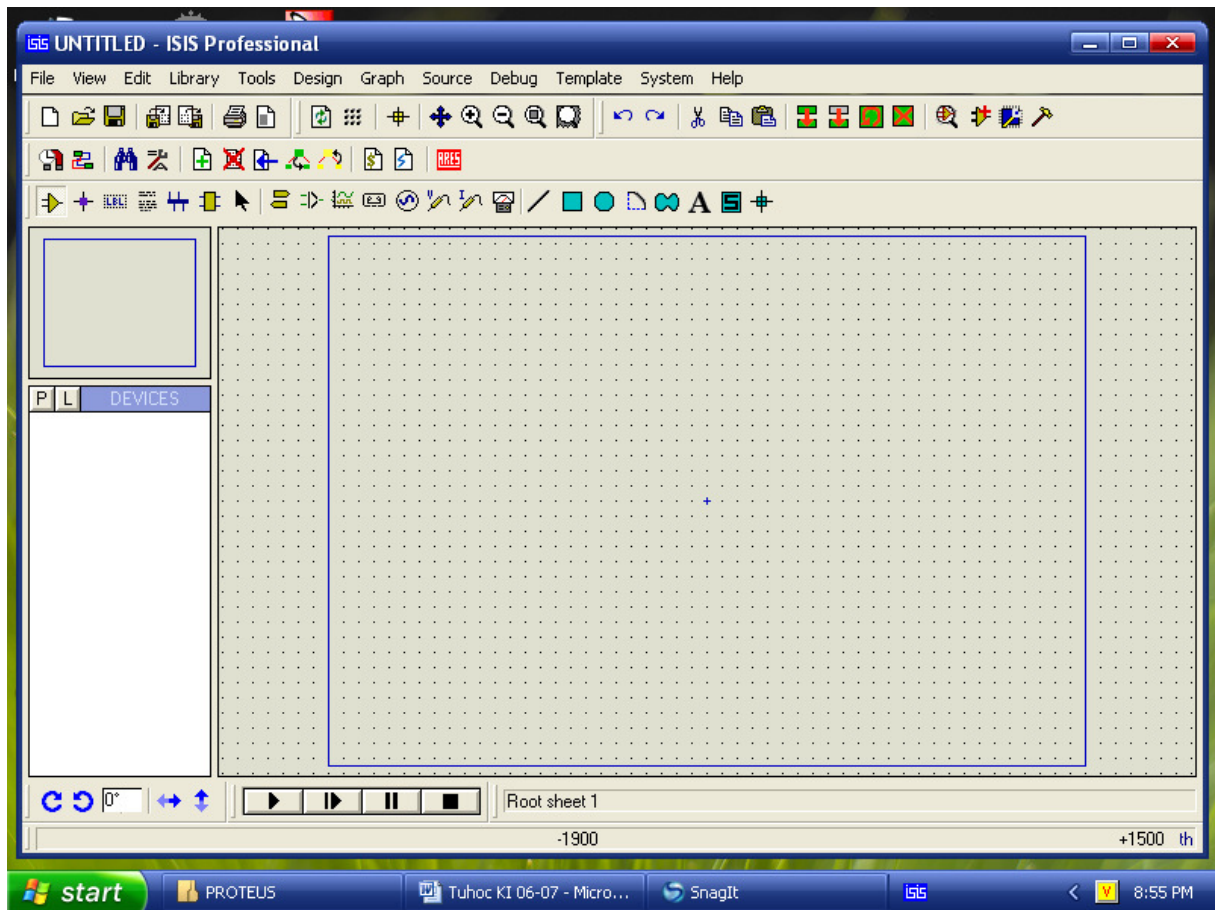
- Nháy đúp chuột vào biểu tượng trên màn hình Desktop.




Cửa sổ thể hiện việc khởi động chương trình:



Sau khi chương trình được khởi động một giao diện được mở ra như hình dưới và việc tìm hiểu chức năng của từng đối tượng tác giả sẽ trình bày ở chương sau.

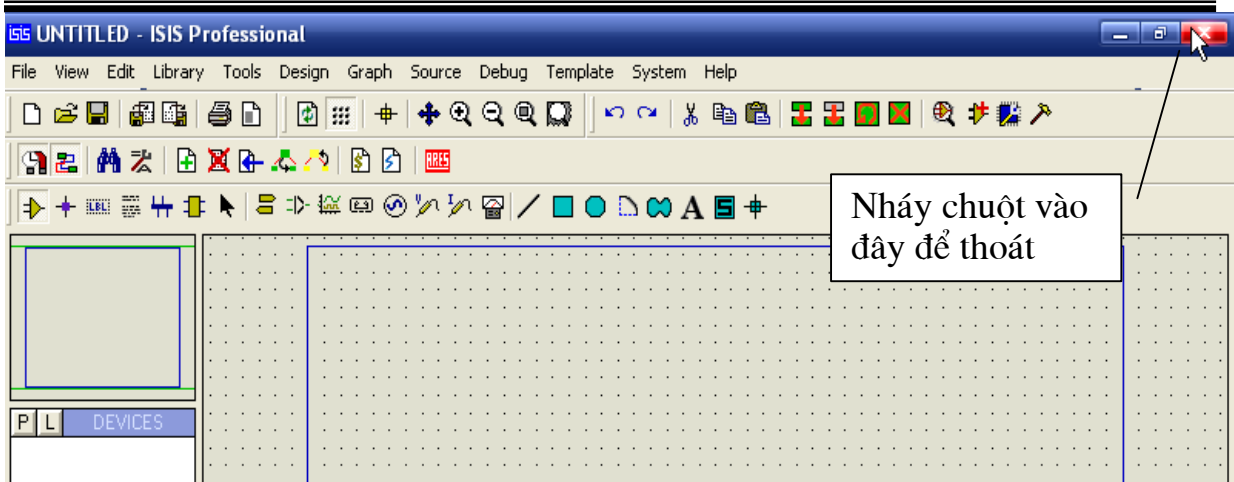


* Khi bắt tay vào thiết kế bạn luôn luôn phải thao tác lưu File lại bằng cách vào File\Save; hoặc nháy vào nút  trên thanh công cụ; hoặc sử dụng tổ hợp phím nóng 'Ctrl+S' trên bàn phím:
Mặc định của phần mềm là khi thực hiện ghi lần đầu tiên thì ghi vào thư mục BIN (nằm trong ổ C), bạn hãy chọn lại đường dẫn đến nơi khác trong ổ cứng của mình.

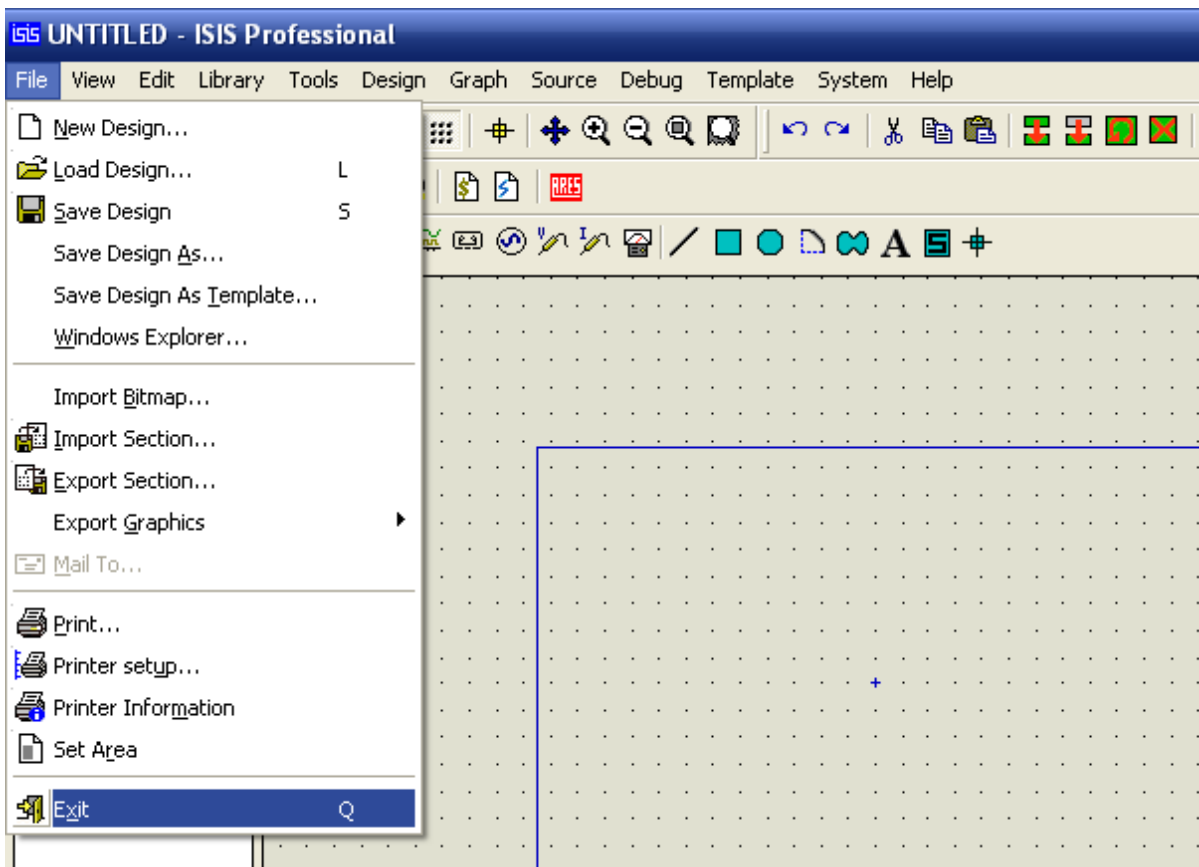
b. Thoát khỏi chương trình.

Thao tác thoát khỏi chương trình được thực hiện khi người thiết kế không thao tác trên phần mềm nữa. Thao tác này được tiến hành bằng một trong các cách cơ bản sau:

- Nháy trái chuột vào nút **Close** phía trên góc bên phải màn hình.
- Nhấn trực tiếp phím Q trên bàn phím (nên làm theo cách này).
- ấn tổ hợp phím 'Alt+X' trên bàn phím.



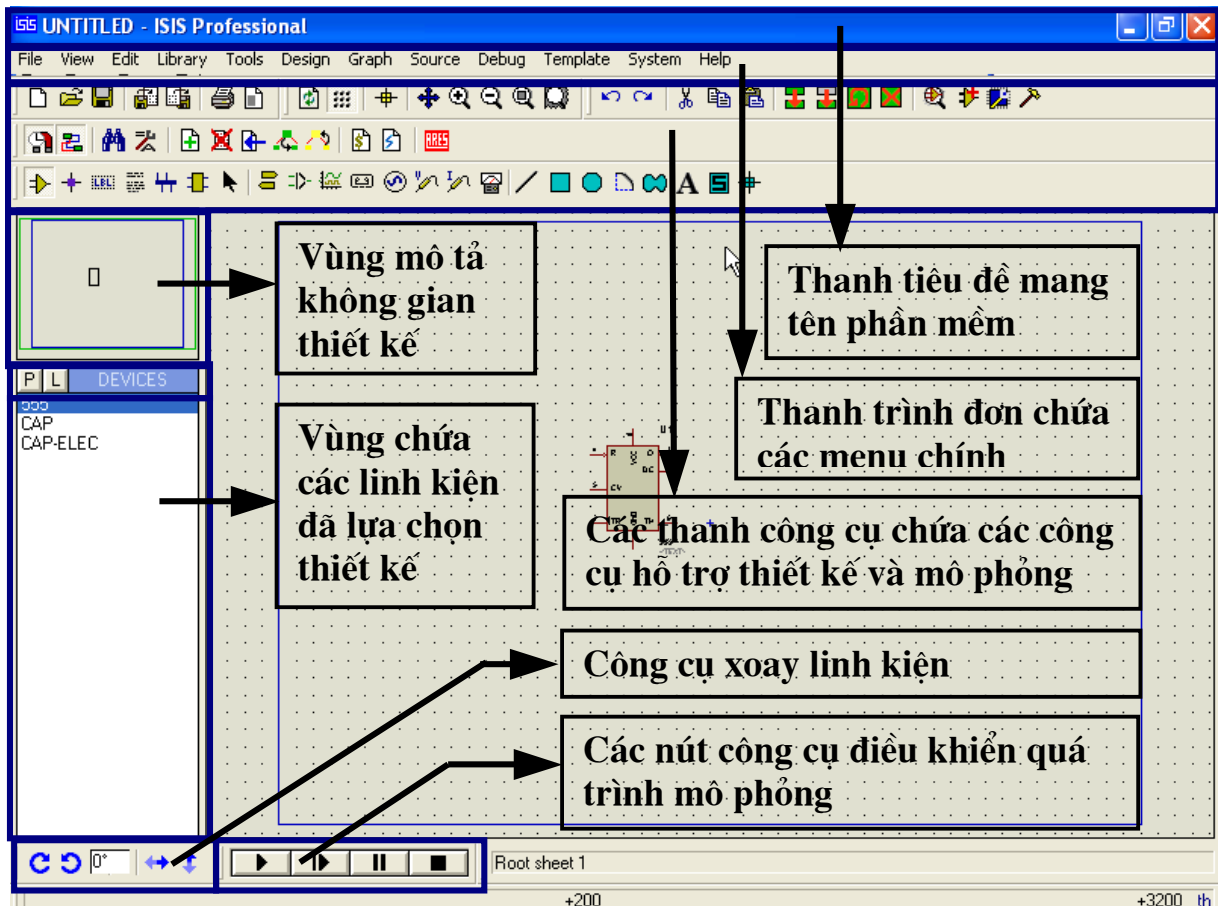
- Chọn File \ Exit.



CHƯƠNG 2: GIỚI THIỆU GIAO DIỆN PHẦN MỀM

2.1. Giới thiệu chung

Sau khi cài đặt hoàn tất phần mềm, để bắt đầu làm quen và sử dụng được phần mềm thì đầu tiên ta khởi động chương trình theo các cách đã nêu ở trên. Chương trình sau khi được khởi động có giao diện như sau:



2.2. Thứ tự xếp đặt trên màn hình (Screen layout).

* Trên cùng là **thanh tiêu đề** mang tên phần mềm. Bên phải có 3 biểu tượng để thu nhỏ, phóng to và thoát chương trình.



* Tiếp đến là **thanh trình đơn (The Menu Bars)** chứa các trình đơn chính tương tự như các phần mềm ứng dụng khác của Windows. Các mục trong thanh trình đơn có chức năng hỗ trợ việc thiết kế và mô phỏng mạch điện.



* Tiếp theo nữa là *các thanh công cụ (The Toolbars)*. Để thay vì mỗi lần tác động lại phải vào trong các menu tương ứng thì phần mềm đưa ra các nút công cụ đặt trên các thanh công cụ cho phép người thiết kế thao tác nhanh hơn. Chức năng từng nút công cụ ta sẽ khảo sát ở phần sau và đồng thời làm quen dần dần trong khi thiết kế. Và thứ tự các thanh này được xếp đặt như sau:

- *Các thanh công cụ lệnh (Command Toolbars)*: Theo mặc định của phần mềm các thanh này được xếp đặt trên màn hình như sau:

Thanh chứa các lệnh File/Print:



Thanh chứa các lệnh hỗ trợ hiển thị (Display Commands):



- Các nút phóng to, thu nhỏ
- Ẩn hoặc hiện lưới màn hình
- Refresh màn hình

Thanh chứa các lệnh hỗ trợ soạn thảo (Editing Commands):



- Xoá đối tượng.
- Xoay đối tượng.
- Di chuyển đối tượng.
- Sao chép đối tượng.
- Dán đối tượng.
- Sao chép nhóm đối tượng.
- Cắt đối tượng.
- Redo và Undo.

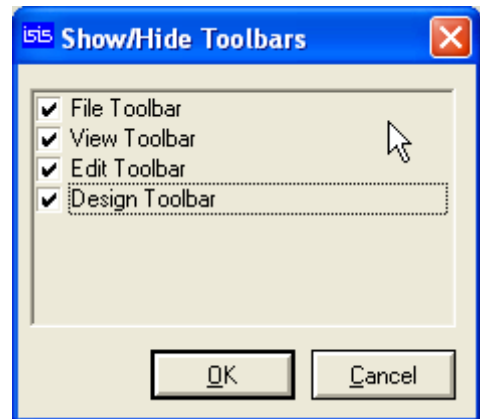
Thanh chứa các công cụ hỗ trợ việc thiết kế (Design Tools):



- Chuyển sang mạch in.

Note: Nếu bạn làm việc với một màn hình Monitor nhỏ bạn có thể giấu bất cứ hoặc tất cả các thanh công cụ trên đi bằng cách chọn Menu View\ Toolbars sau đó thiết lập theo mong muốn.

- **Thanh công cụ chọn lựa chế độ (Mode Selector Toolbar):** Vị trí thanh này đặt ở bên dưới phía tay trái của màn hình hiển thị dùng để chọn lựa các chế độ trong lúc soạn thảo; chúng được tác động ngẫu nhiên khi bạn kích chuột lên.

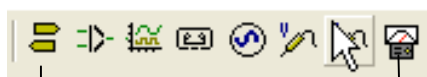


Thanh chọn các chế độ chính (Main Modes):



- Vẽ đường BUS
- Viết văn bản
- Viết nhãn (chú thích)
- Hiển thị điểm nối
- Trở lại cửa sổ linh kiện

Thanh chứa các thiết bị và dụng cụ (Gadgets):



- Lấy các thiết bị đo : máy hiện sóng, Vol met (AC,DC), Ampemet (AC, DC).
- Lấy nguồn cung cấp Vcc, GND

Thanh hỗ trợ vẽ hình đồ họa không gian 2 chiều (2D Graphics):



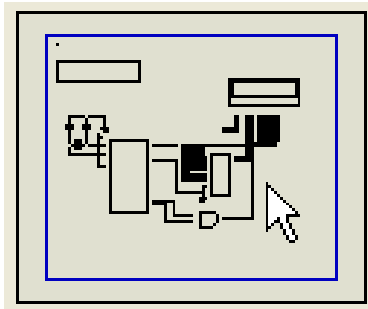
Note: Các thanh này luôn luôn được hiển thị (không thể ẩn đi được như các thanh công cụ lệnh) và các chức năng của nó là duy nhất vì không được bố trí đồng thời trên các thanh trình đơn.

- **Thanh công cụ hỗ trợ định hướng đối tượng (Orientation Toolbar):** Trên thanh này chứa các nút có chức năng làm xoay và lật ngược (rotation and reflection) các đối tượng đã được lựa chọn:



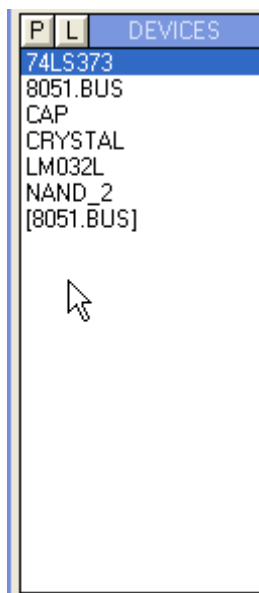
- ➔ Lật đối tượng theo chiều dọc
- ➔ Lật đối tượng theo chiều ngang
- ➔ Hiển thị góc xoay (0° , $90^\circ/-90^\circ$, $180^\circ/-180^\circ$, $270^\circ/-270^\circ$)
- ➔ Xoay đối tượng 1 góc 90° theo chiều mũi tên (ngược)
- ➔ Xoay đối tượng 1 góc 90° theo chiều mũi tên (thuận)

- Cửa sổ tổng quan (The overview Window):



Đây là cửa sổ hiển thị nguyên vẹn vùng nội dung bản thiết kế trong một khung lưới và bạn có thể làm thay đổi tọa độ của khung lưới này bằng cách click chuột trái lên một vị trí bất kì trên khung này.

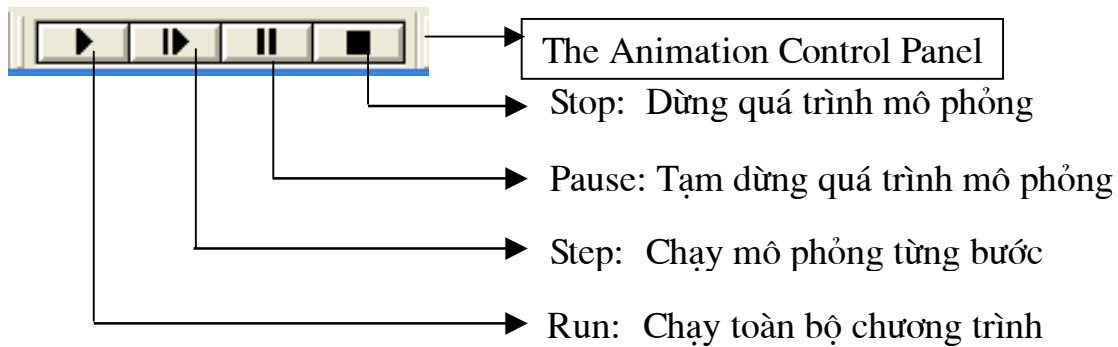
- Cửa sổ chọn đối tượng (The Object Selector):



Sau khi vào thư viện linh kiện tìm và lấy ra các linh kiện theo yêu cầu của mạch thì các linh kiện khi đã được chọn sẽ hiển thị trong cửa sổ này (ví dụ trong mạch cần IC74LS373, tụ điện CAP, thanh anh CRYSTAL .. thì các linh kiện này khi được chọn ra từ thư viện linh kiện sẽ hiển thị trong cửa sổ này như hình bên). Khi các linh kiện đã được liệt kê hết ra rồi, trong quá trình thi công mạch cần lấy linh kiện nào ra vùng làm việc thì ta nháy chuột vào linh kiện đó để lựa chọn (như hình bên thì linh kiện 74LS373 đang được lựa chọn để lấy ra thiết kế).

- Bảng điều khiển mô phỏng (The Animation Control Panel)

Cũng giống các thiết bị như VCD, Radio casset để điều khiển hoạt động của máy người ta thiết kế một bảng điều khiển với nhiều nút chức năng thì trong phần mềm này để điều khiển quá trình mô phỏng các nhà lập trình cũng lập trình đưa ra một bảng điều khiển với các ký hiệu giống như các ký hiệu có trong thực tế:



2.3. Các thao tác cơ bản hay sử dụng

- Nháy chuột phải để chọn đối tượng (đây là điểm khác cơ bản nhất so với các phần mềm khác).
- Nháy chuột phải liên tiếp hai lần để xoá đối tượng.
- Lăn con lăn trên thân chuột để phóng to hay thu nhỏ vùng làm việc.
- Nhấn phím P để vào thư viện của chương trình.
- Bắt đầu quá trình vẽ ấn phím W.
- Để chạy chương trình ấn Ctrl+F12
- Ấn F6 để phóng to vùng làm việc.
- Ấn F7 để thu nhỏ vùng làm việc.
- Ấn F8 để thu cả trang làm việc về khắp màn hình.
- Ấn G để hiển thị lưới hoặc ẩn lưới.
- Ấn phím X để đầu con trỏ chuột xuất hiện dấu x, hoặc xuất hiện hai đường chỉ dài khắp màn hình (để dễ kết nối).
- Ấn tổ hợp phím Ctrl+Z để quay lại thao tác trước đó.

CHƯƠNG 3:

THỰC HÀNH THIẾT KẾ VÀ MÔ PHỎNG MẠCH ĐIỆN

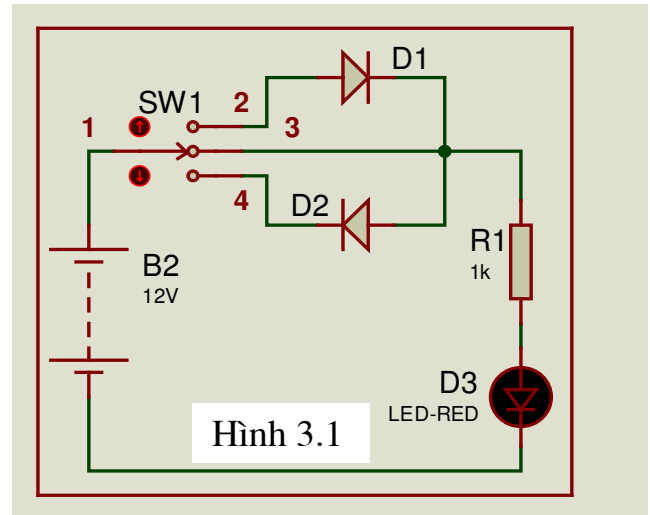
3.1. Thiết kế và mô phỏng mạch tương tự

3.1.1. Mô phỏng hoạt động của Diode

Mô tả mạch điện: Mạch hình 3.1 là Mạch khảo sát hoạt động của 2 Diode D1 và D2, nguồn sử dụng là nguồn Một chiều 12V, một chuyển mạch 3 trạng thái để thay đổi vị trí làm việc của mạch.

Hoạt động của mạch như sau:

- Khi SW1 có 1 đóng với 2:
+ Thì D1 phân cực thuận --> D1 dẫn --> đèn Led hiển thị D3 sáng.
+ D2 hở mạch nên không dẫn
- Khi SW1 có 1 đóng với 3: đèn Led hiển thị D3 sáng (R1 hạn dòng cho D3).
- Khi SW1 có 1 đóng với 4:
+ Thì D2 phân cực thuận --> D2 dẫn --> đèn Led hiển thị D3 sáng.
+ D1 hở mạch nên không dẫn.



Các bước vẽ mạch điện:

Bước 1: Lấy linh kiện

Các linh kiện cần cho mạch gồm có nguồn 1 chiều 12V, Diode, SW, điện trở, Led; tất cả đều nằm trong thư viện của chương trình.

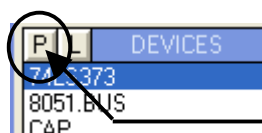
Để vào thư viện linh kiện có thể thực hiện theo 4 cách sau:

- Trên thanh chứa các lệnh hỗ trợ soạn thảo (*Editing Commands*) nháy chuột chọn biểu tượng **Pick devices**



Nháy chuột vào đây

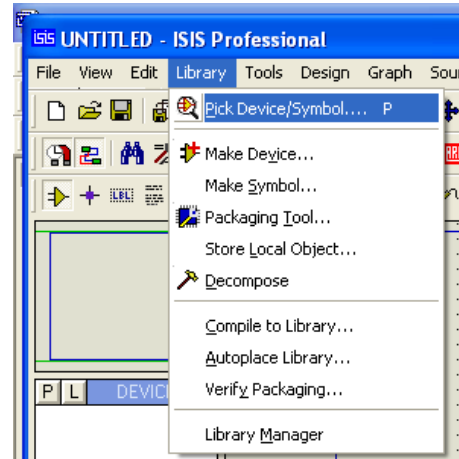
- Trên cửa sổ chọn đối tượng (*The Object Selector*): nháy chuột vào chữ P.



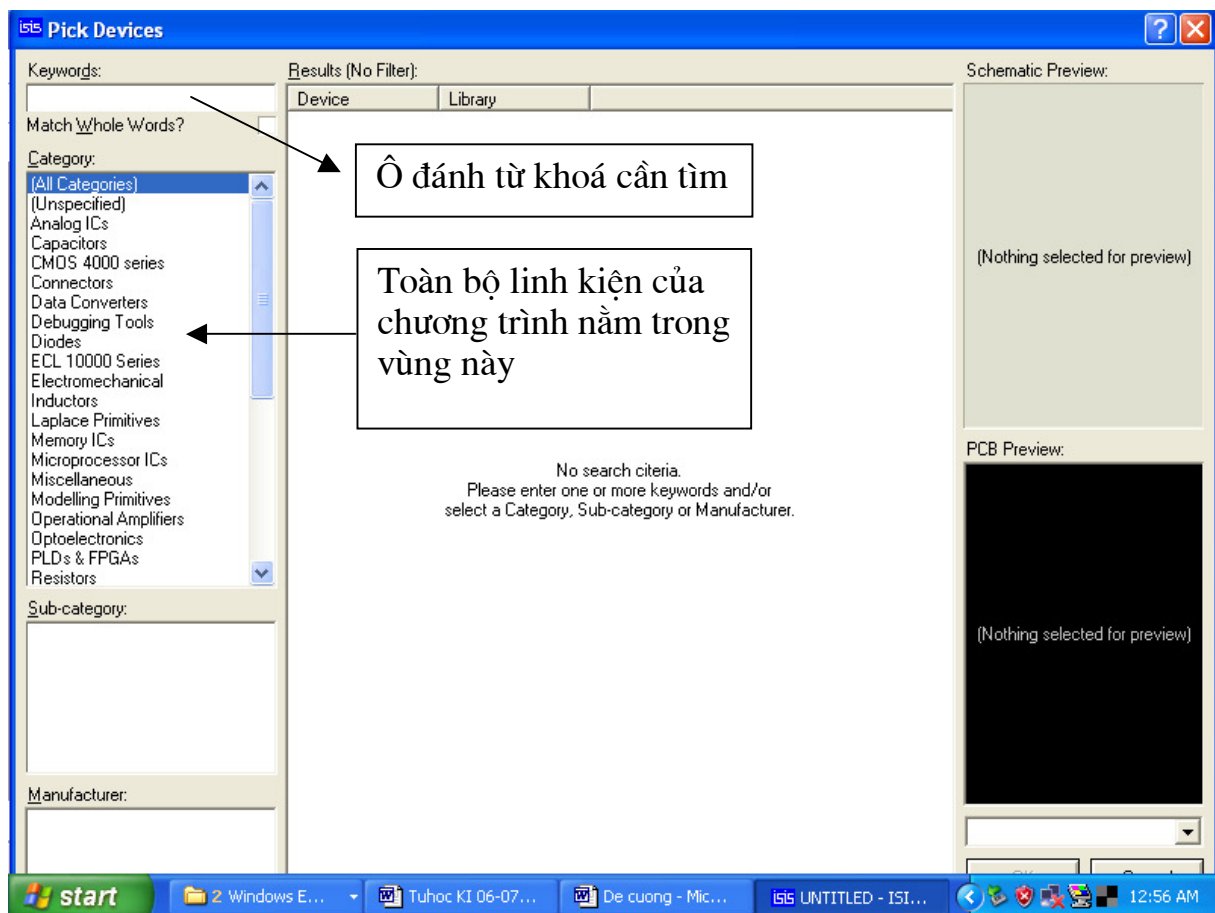
Nháy chuột vào đây

- Trên thanh trình đơn (The Menu Bars) chọn Libray\ Pick Devices/ symbol.....P.

- Ấn trực tiếp phím “p” trên bàn phím (nên làm theo cách này) vì rất đơn giản.

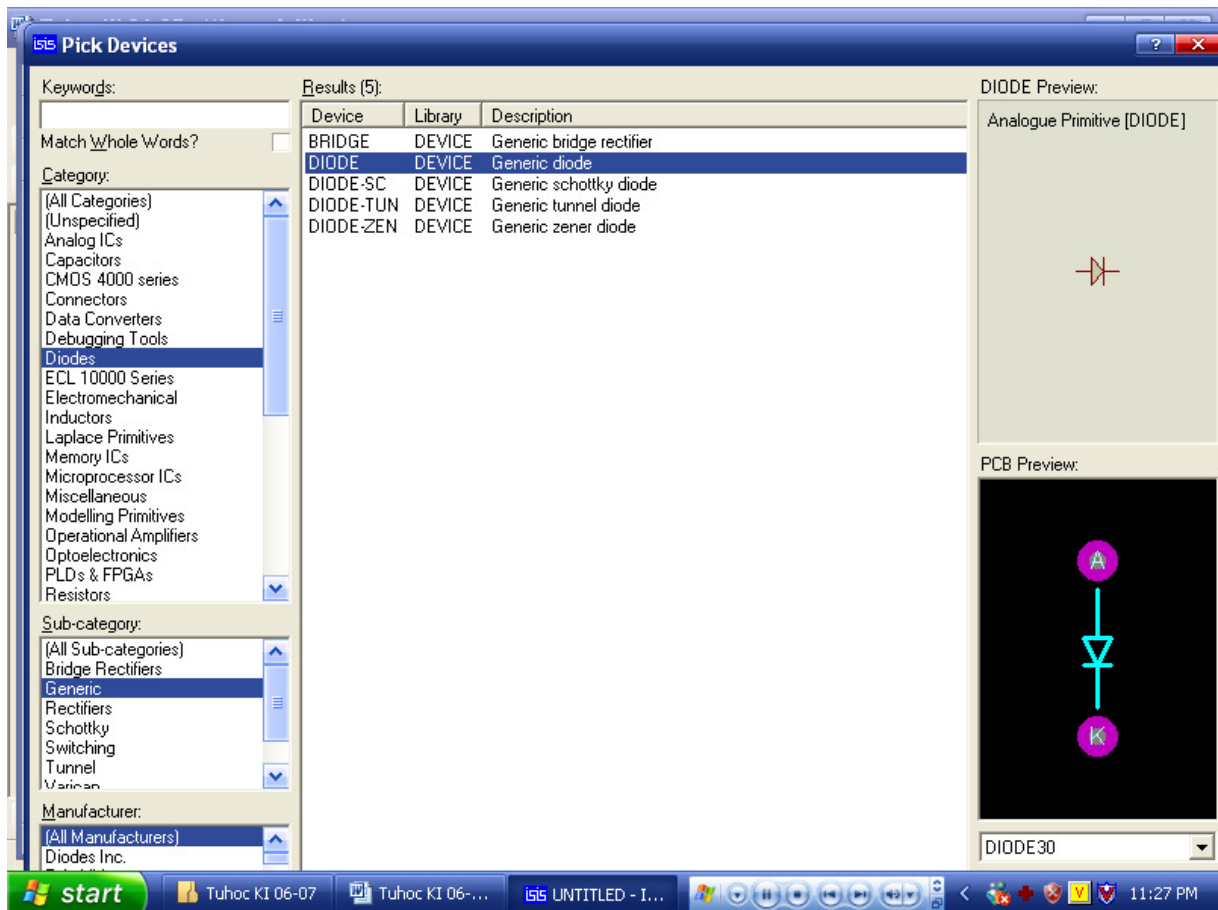


Sau khi truy cập vào thư viện linh kiện thì màn hình như sau:



- Để lấy linh kiện ra: Bạn có thể vào trực tiếp thư mục chứa đối tượng để lấy chúng ra; ví dụ để lấy D1 và D2 bạn nháy chuột vào dòng Diodes --> trong ô Sub-category chọn dòng Generic --> trong ô Description chọn dòng DIODE DEVICE Generic Diode (nháy đúp chuột để chọn).

Hai ô: DIODE Preview (hiển thị kí hiệu của linh kiện được chọn); PCB Preview hiển thị sơ đồ chân linh kiện (hỗ trợ vẽ mạch in PCB). Hình dưới.



Tương tự như vậy:

Để lấy SW bạn vào Switches & Relay --> SW-ROT-3.

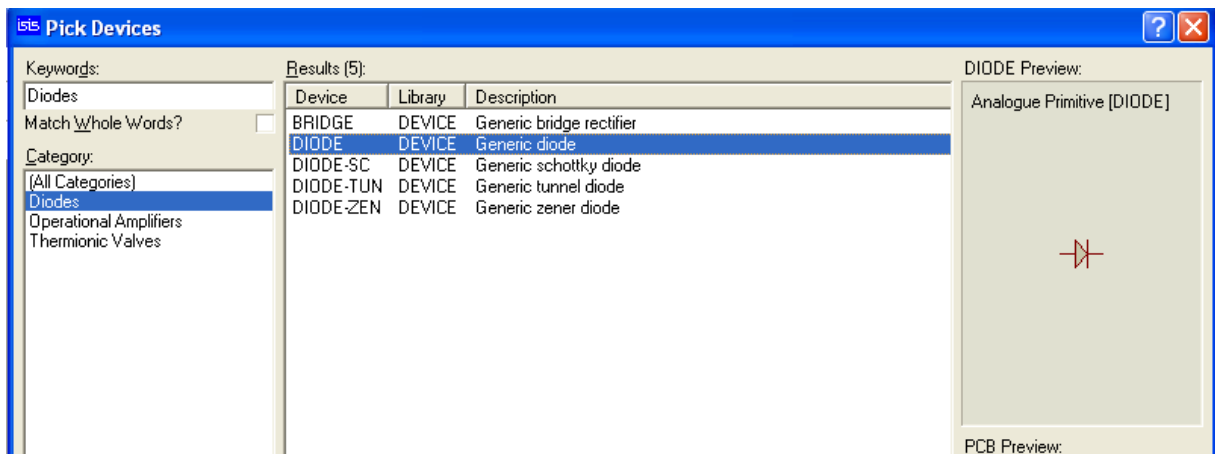
Để lấy nguồn 12V bạn vào Miscellaneous--> BATTERY.

Để lấy R1 bạn vào Resistors --> Generic --> RES.

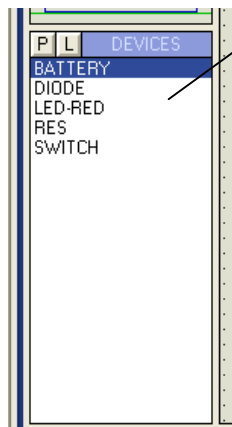
Để lấy D3 (LED-RED) bạn vào Optoelectronics --> LED-RED.

Ngoài ra bạn còn có thể lấy các linh kiện mà mạch yêu cầu bằng cách trong ô Keywords ta nhập tên các linh kiện cần lấy rồi nhấn Enter. Sau đó để chọn linh kiện ta cũng làm tương tự như trên.

Ví dụ cần lấy Diode trong ô Keywords bạn nhập từ khoá “Diode” rồi Enter.



Khi các linh kiện cần thiết được chọn thì chúng nằm thường trực trong vùng dưới, và khi cần đưa ra vùng làm việc ta không phải vào thư viện lấy ra nữa mà sử dụng luôn những linh kiện này nhiều lần.

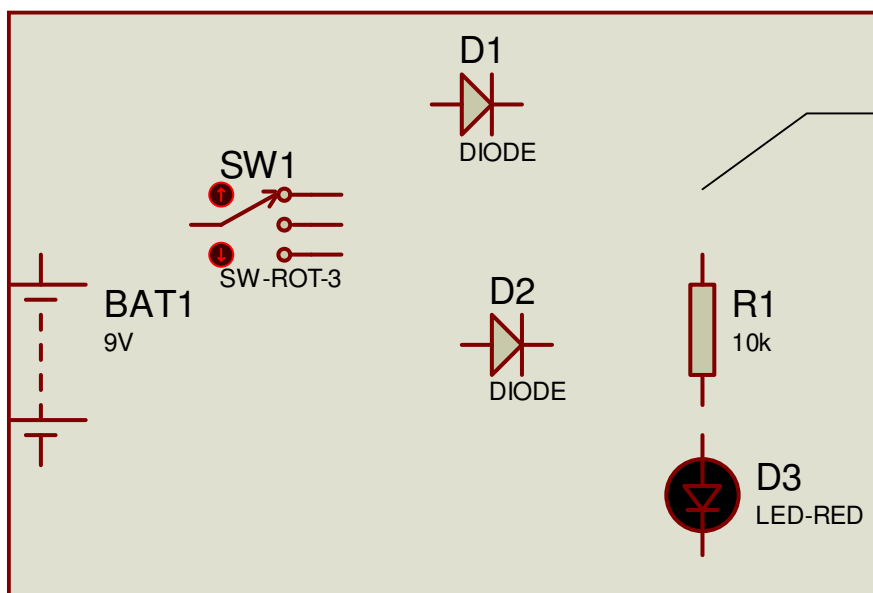


Vùng chứa các linh kiện sau khi đã được chọn

Để đưa các linh kiện ra vùng làm việc ta nhấn chọn linh kiện sau đó di chuyển chuột ra vùng làm việc rồi nhấn trái chuột. Trong quá trình này bạn có thể sử dụng phóng to thu nhỏ vùng làm việc bằng cách lăn con xoay trên chuột. Chỉ đơn giản như vậy là bạn có thể lấy linh kiện ra được rồi đấy. Bạn hãy thử làm nhé, chúc bạn thành công.

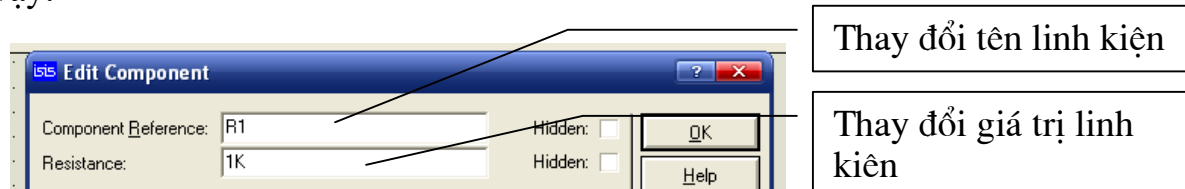
Bước 2: Sắp xếp và đặt tên linh kiện

- Để thực hiện bước này nhanh chóng bạn nên làm quen với các công cụ xoay linh kiện , công cụ di chuyển (move) một hoặc một nhóm linh kiện, công cụ xóa (delete) và sao chép (copy) linh kiện.
- Khi cần xoay linh kiện bạn làm như sau: Nháy chuột phải vào linh kiện cần xoay (linh kiện chuyển thành màu đỏ) sau đó nháy trái chuột vào nút công cụ xoay để xoay cho phù hợp.
- Khi cần di chuyển linh kiện hoặc một nhóm linh kiện thì đầu tiên bạn cũng nháy chuột phải để chọn linh kiện hoặc một nhóm linh kiện --> tiếp tục nháy trái chuột vào nút Move Tagged Objects --> sau đó di chuột tới vị trí mới cần đặt rồi nháy trái chuột.



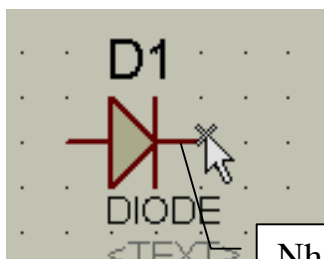
Linh kiện sau khi sắp xếp.

- Khi cần xóa linh kiện hay một đối tượng cách đơn giản nhất là nháy hai lần chuột phải lên đối tượng cần xóa --> đối tượng sẽ được xóa.
- Để đặt tên linh kiện: ví dụ thay trị số cho R1 trước tiên nháy chuột phải vào R1 (R1 chuyển sang màu đỏ) --> nháy chuột trái vào R1 --> cửa sổ Edit Component hiện ra bạn thay đổi thứ tự điện trở trong ô Component Reference và giá trị của điện trở trong ô Resistance rồi chọn OK. Các linh kiện khác cũng tương tự như vậy.

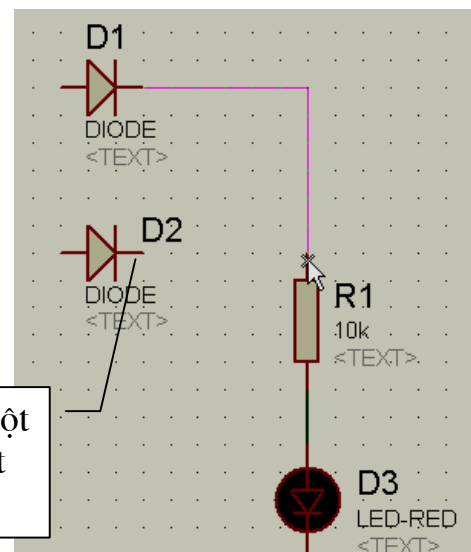


Bước 3: Kết nối mạch điện

Thực hiện kết nối các linh kiện lại với nhau theo sơ đồ nguyên lý bạn làm như sau: Di chuyển mũi tên (chuột) tới chân linh kiện chuẩn bị nối --> khi đầu mũi tên xuất hiện dấu x thì chân đó được chấp nhận --> nháy chuột trái --> kéo rê đến chân linh kiện cần kết nối --> khi mũi tên lại xuất hiện chữ x --> nháy chuột trái tiếp để kết thúc quá trình vẽ mạch. Vẽ các đường khác cũng làm tương tự như vậy.



Nháy chuột trái để bắt đầu.



Nháy chuột trái để kết thúc.

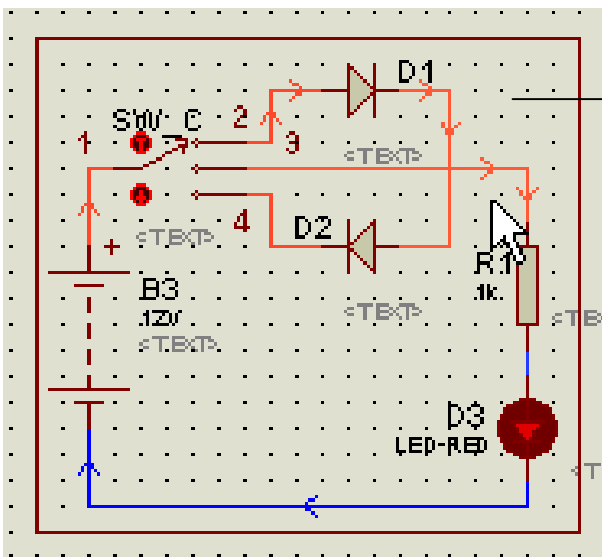
Bước 4: Mô phỏng mạch điện

Sau khi mạch điện đã được kết nối xong việc tiếp theo là cho chạy mô phỏng để xem hoạt động của mạch. Các công cụ hỗ trợ mô phỏng đó là công cụ RUN,



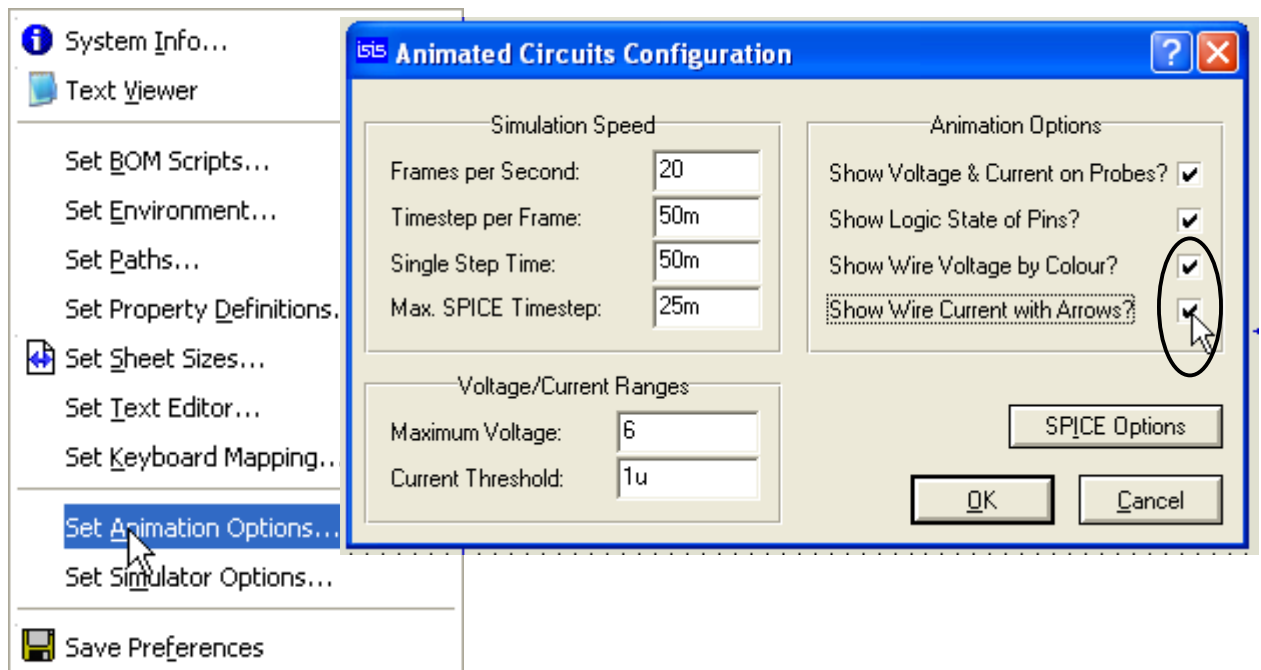
- Để bắt đầu quá trình mô phỏng nháy chuột vào nút RUN (hình tam giác) hoặc sử dụng tổ hợp phím nóng Ctrl+F12.
- Khi chương trình chạy hình tam giác đen chuyển thành màu xanh.
- Dùng chuột tác động vào SW rồi quan sát hiện trạng của bóng đèn Led:

- Để tăng tính trực quan, bạn có thể cho hiển thị chiều của dòng điện như sau:
(dùng cho các bài giảng khi cần mô phỏng hoạt động của linh kiện).



Chiều đi của dòng điện (chiều mũi tên).

Để làm được như vậy đầu tiên bạn vào System --> chọn Set Animation Options -> đánh dấu chọn hai box: Show Wire Current with Arrows? Và Show Wire Voltage by Colour?

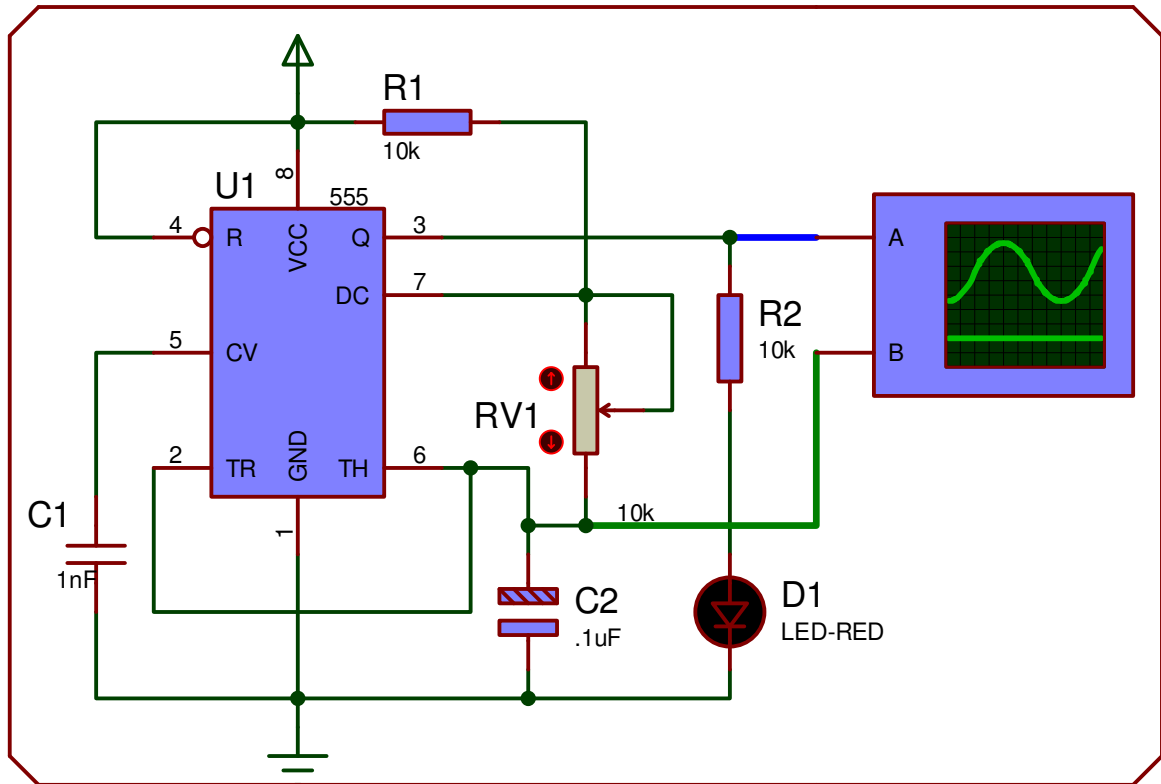


3.1.2. Mạch tạo dao động dùng IC 555.

Yêu cầu:

- Vẽ mạch tạo dao động tạo xung dùng IC555 như sau:
- Chạy mô phỏng mạch điện:
- Sử dụng các thiết bị đo kiểm tra của phần mềm đo kiểm tra dạng sóng ngõ ra (chân 3) và dạng sóng trên tụ (Chân 6).

- Cân chỉnh trị số của các linh kiện để dạng sóng ngõ ra có tần số là $f = 1\text{KHz.}, 5\text{KHz}, 10\text{KHz.}$



Các bước thực hiện:

Bước 1: Lấy linh kiện

- Để lấy các linh kiện như IC555, R, C, LED ta truy cập vào thư viện linh kiện của chương trình bằng một trong các cách sau:

+ Nháy chuột vào nút công cụ Pick Devices trên thanh công cụ



Click vào đây

+ Trên cửa sổ chọn đối tượng nháy chuột vào công cụ chữ P :



Click vào đây

+ Ấn phím nóng 'p' trên bàn phím.

Sau khi thư viện linh kiện mở ra: Nếu có thời gian bạn nên tìm hiểu vị trí lưu trữ của từng linh kiện để việc lấy linh kiện mất ít thời gian nhất.

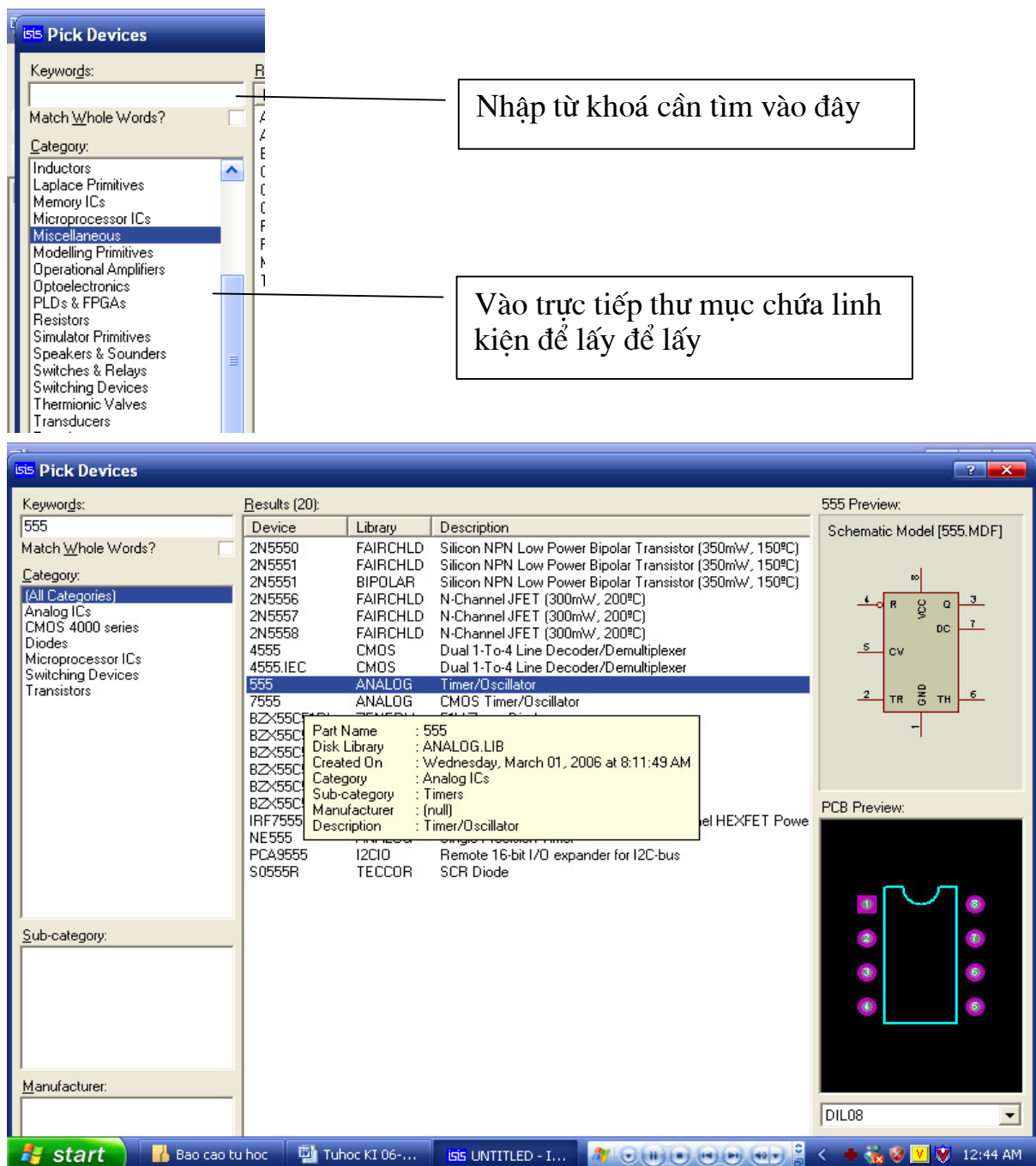
Nói chung các linh kiện được sắp xếp theo thứ tự A, B --> Z và tên các linh kiện được thể hiện bằng thuật ngữ tiếng anh: Ví dụ Tụ điện (Capacitors) ,

điện trở (Resistors), cuộn dây (Inductors), Diốt (Diodes), Transistor (Transistors) , phím bấm (Button) ...

- Cách lấy linh kiện ra như sau: Bạn có thể vào trực tiếp thư mục chứa đối tượng để lấy chúng ra hoặc trong Textbox Keywords nhập từ khoá cần tìm rồi nhấn Enter.

Trong bài tập này theo yêu cầu của hình ta cần IC 555, R, C, VR, LED.

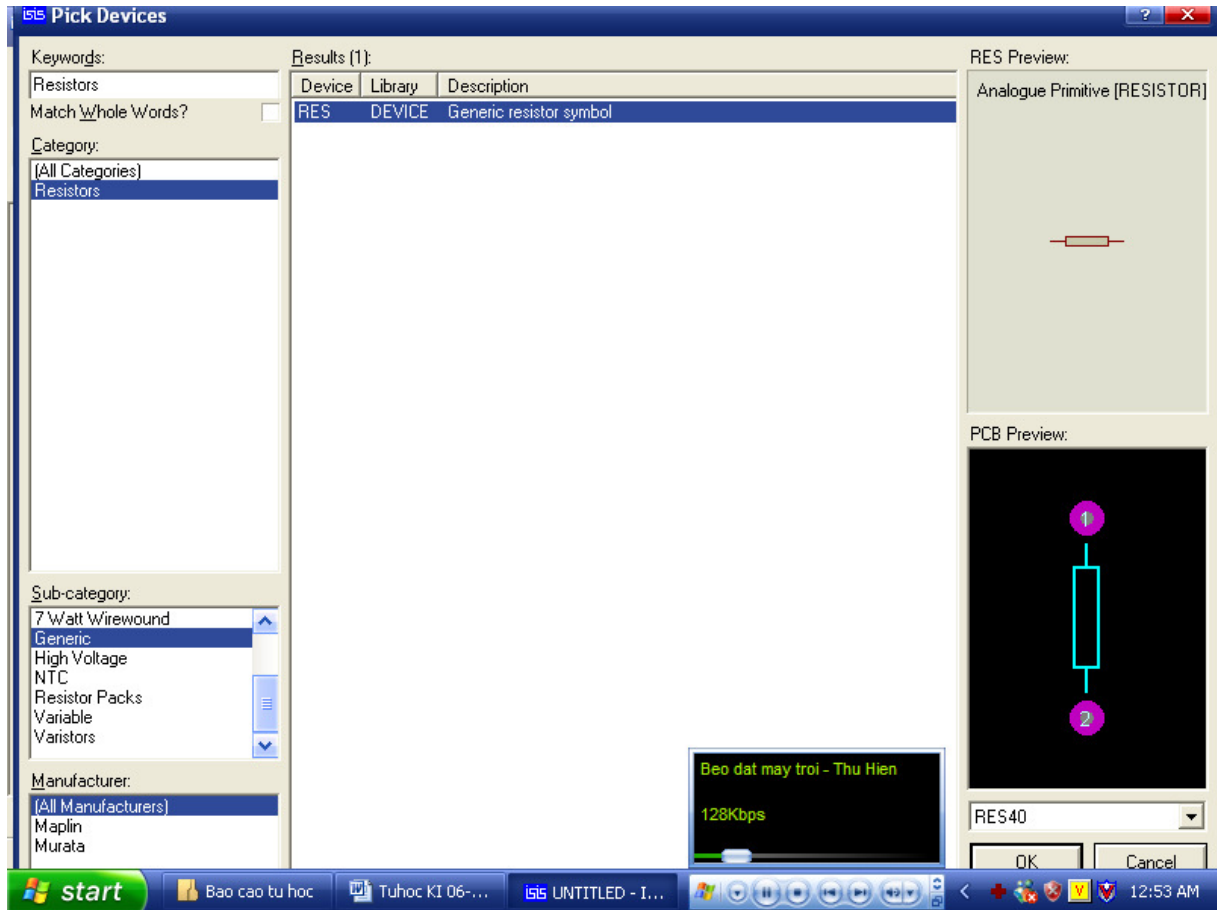
Để lấy IC 555 trong ô Keywords ta nhập từ khóa: “555” rồi nhấn Enter khi quá trình tìm kiếm kết thúc thì tất cả các linh kiện liên quan tới từ khóa “555” đều được hiển thị ra. Bạn hãy chọn IC 555 mà mình cần bằng cách nhấp đúp chuột trái vào dòng 555 ANALOG Timer/oscillator (hình vẽ).



Nhập từ khoá cần tìm vào đây

Vào trực tiếp thư mục chứa linh kiện để lấy

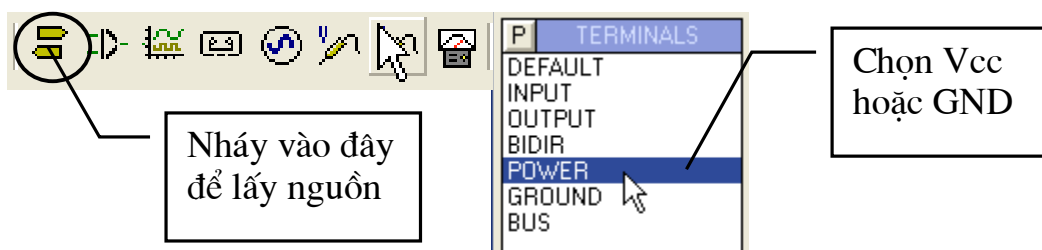
Tương tự để lấy điện trở bạn nhập từ khoá Resistors rồi nhấn Enter, tất cả các linh kiện điện trở được tìm ra không nên lấy tùy ý, nếu cần điện trở có kích thước chuẩn để thiết kế mạch in thì bạn tìm và chọn cho chính xác còn không thì bạn chọn theo đường dẫn sau: trong vùng Sub-category chọn mục Generic sau đó thao tác như trên để lấy ra.




- Để lấy biến trở: Trong ô Keywords nhập Resistors rồi Enter --> trong Sub-category chọn Variable --> trong mục Device chọn POT-LIN ACTIVE.

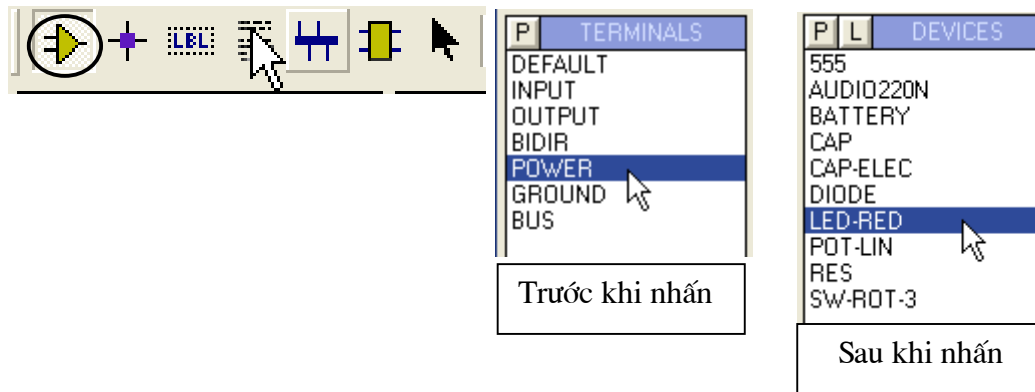
- Để lấy tụ điện: Trong ô Key words nhập Capacitors rồi Enter --> trong Sub-category chọn Generic --> trong mục Device chọn CAP DEVICE (nếu cần tụ không phân cực) hoặc chọn CAP-ELEC DEVICE (nếu cần tụ phân cực).

- Để lấy nguồn cấp bạn nháy vào  trên thanh công cụ sau đó chọn POWER và GROUND.

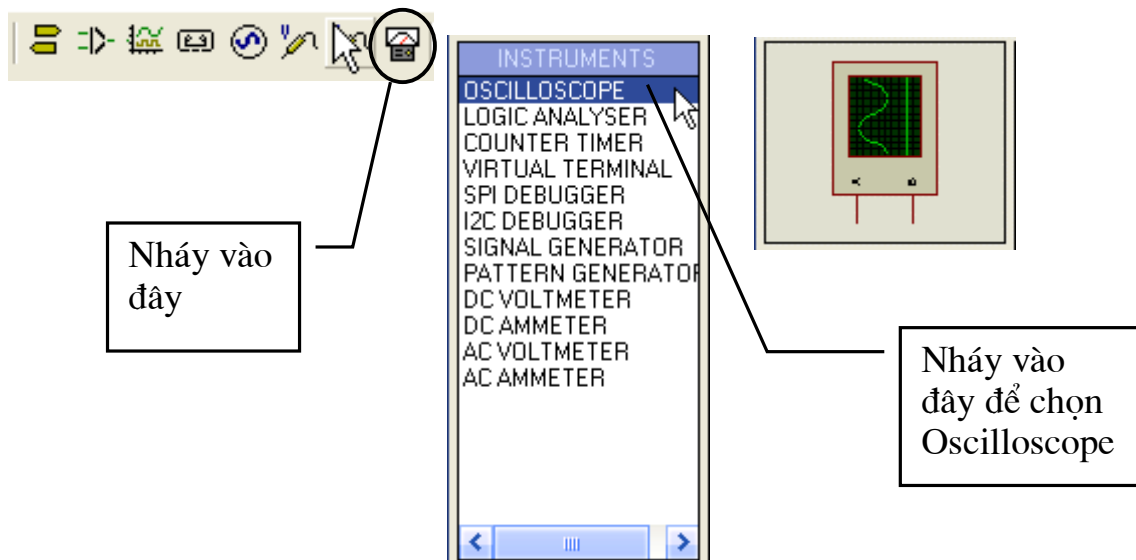


- Sau khi lấy xong nguồn cấp để trở về cửa sổ linh kiện ban đầu bạn nhấn chuột

vào . (Những người mới làm quen thường lúng túng ở chỗ này).



- Để lấy máy hiện sóng bạn nhấn vào nút công cụ rồi chọn Oscilloscope.



Bước 2: Sắp xếp và đặt tên linh kiện

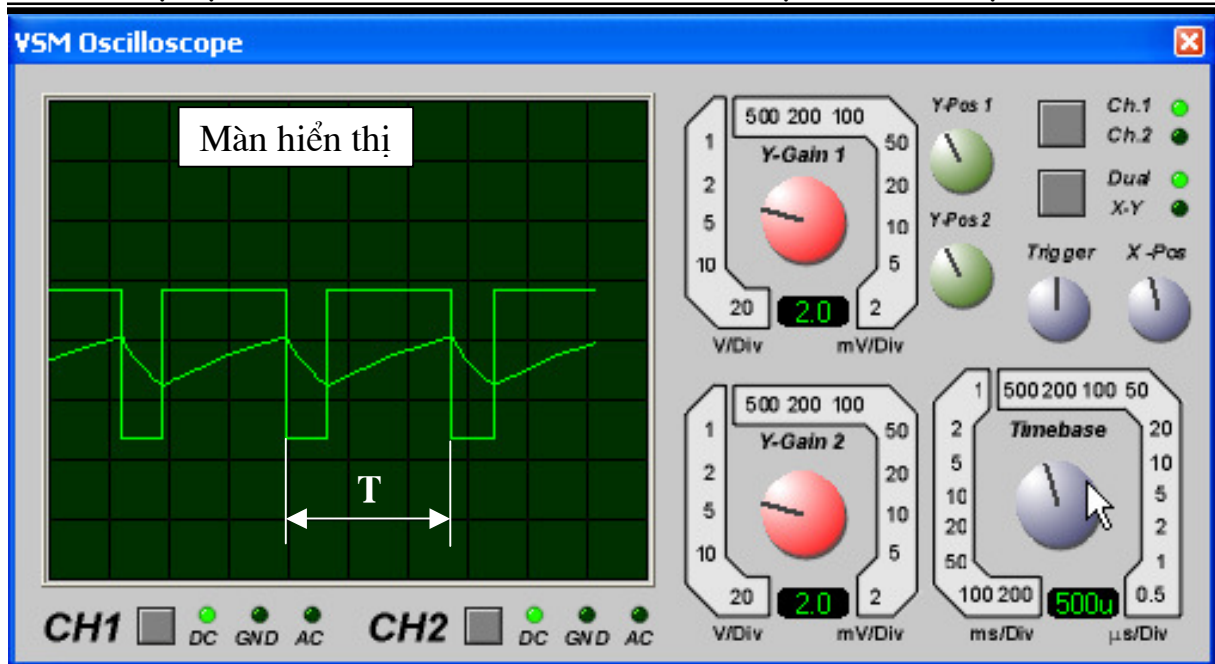
(Bạn thực hiện như phần bước 2 mục 3.1.1).

Bước 3: Kết nối mạch điện

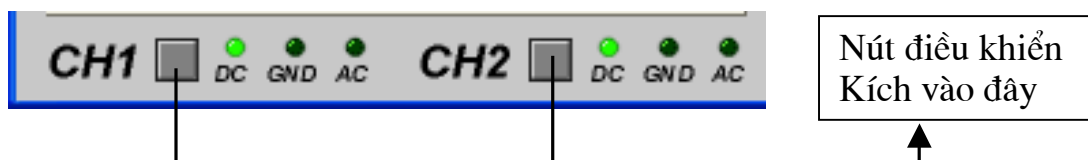
(Bạn thực hiện như phần bước 3 mục 3.1.1).

Bước 3: Mô phỏng mạch điện

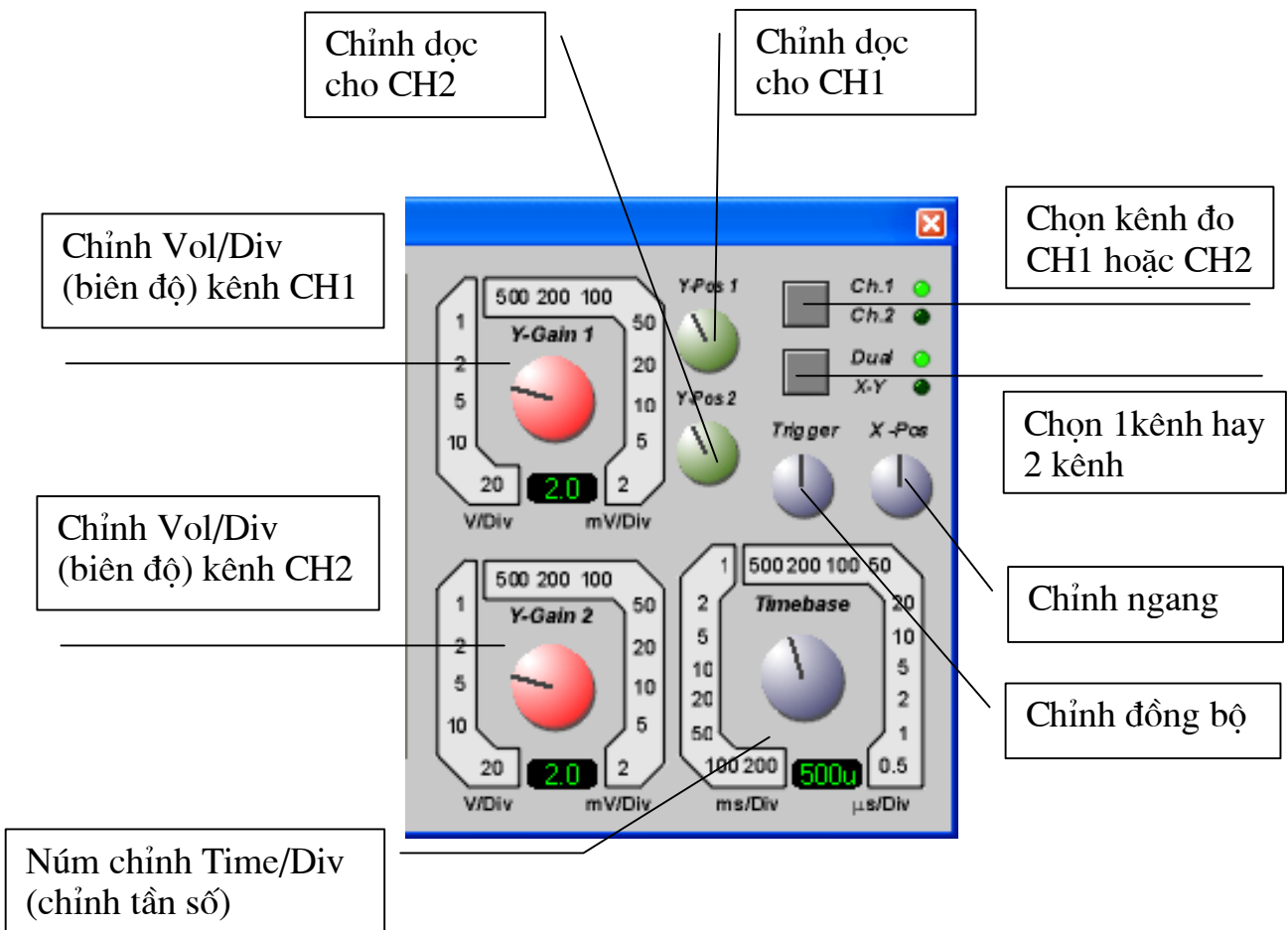
Sau khi sắp xếp và đặt tên, thay đổi giá trị linh kiện như sơ đồ yêu cầu, Bạn nhấn RUN để bắt đầu mô phỏng. Khi nhấn nút RUN chương trình chạy đồng thời cửa sổ hiển thị của Oscilloscope (máy hiện sóng) xuất hiện (việc bố trí các nút chức năng giống hệt như một máy Oscilloscope thực tế (khi sử dụng thành thạo các nút, nút chức năng trên màn này tức là bạn đã một phần sử dụng được Oscilloscope ngoài thực tế), Bạn hãy từ từ làm quen với các phím chức năng để điều khiển cho thích hợp: Đây là máy hiện sóng hai tia (hai ngõ vào CH1 và CH2).



- Hai nút chọn chế độ đo cho hai kênh CH1 và CH2; mặc định là đo tín hiệu DC, khi cần thay đổi sang AC hoặc GND bạn kích trực tiếp lên nút điều khiển.



- Chức năng các nút trên bảng điều khiển chính:



- Chỉnh RV1 và quan sát dạng sóng ngõ ra --> thấy chu kỳ của sóng thay đổi (tức là tần số của sóng thay đổi):

- Căn cứ vào dạng sóng trên màn hiển thị ta tính được tần số của sóng ngõ ra:

Một chu kỳ sóng là 2,8 ô (số div = 2,8), Time/div = 500us = $500 \cdot 10^{-6}$ s.

$$\Rightarrow \text{Chu kỳ } T = 2,8 \times 500 \cdot 10^{-6} = 1400 \cdot 10^{-6} \text{ (s)}$$

$$\Rightarrow \text{Tần số } f = 1/T = 1/1400 \cdot 10^{-6} = 700 \text{ Hz.}$$

Tương tự ta cũng tính được biên độ (cạnh trên đến cạnh dưới) của dạng sóng như sau: Số div = 2,5 (từ cạnh trên đến cạnh dưới)

Số Vol/div kênh CH1 = 2

==> Biên độ :

$$V_{pp} = 2,5 \times 2 = 5 \text{ (Vpp).}$$

Như vậy để có tần số là 1KHz bạn thực hiện thay đổi lại trị số của linh kiện C2 hoặc RV1 để đạt được yêu cầu:

3.1.3. Bài tập

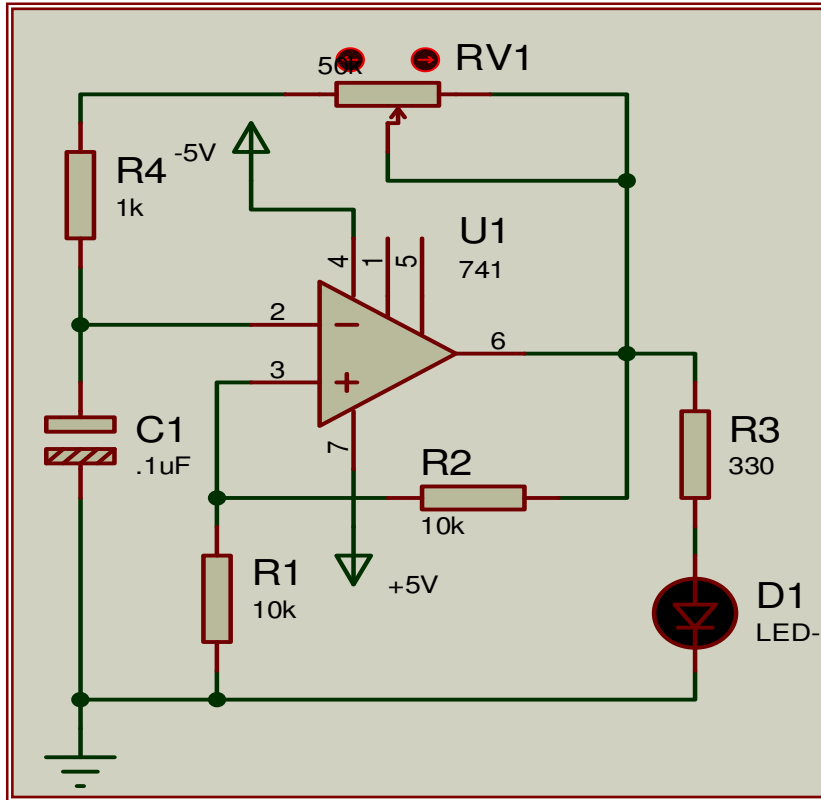
Bài 1:

Hãy vào Help \ Samples Designs \ Interactive Simulation Samples \ Animated Circuits: Khảo sát và vẽ mô phỏng lại các mạch trong các mục sau:

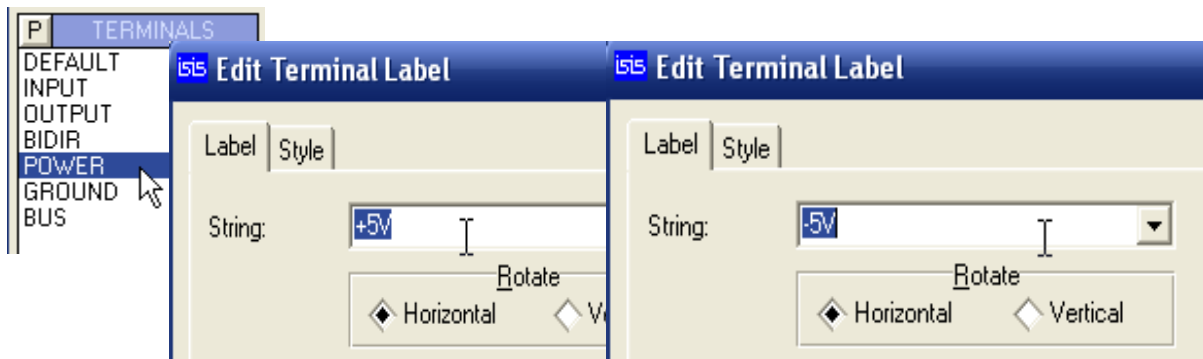
- Capacitors: Circuits 1, 2,3. Khảo sát đặc tính phóng nạp của tụ điện.
- Diodes and Rectification: Circuit 1 đến Circuit 10. Khảo sát hoạt động của Diode.
- Inductors and Relays: Circuits 1, 2, 3. Các mạch giao tiếp với Role.
- Transistors Circuits: Circuit 1 đến Circuit 7. Khảo sát hoạt động của Transistor.
- Oscillators: Các mạch tạo dao động
 - + Circuit 1: Mạch tạo dao động L, C.
 - + Circuit 2: Mạch đa hài dùng 2 Transistors.
 - + Circuit 3: Mạch tạo dao động dùng IC555.

Bài 2:

- Vẽ và mô phỏng mạch tạo xung dùng IC 741 như sau:
- Dùng máy hiện sóng đo dạng sóng tại đầu ra (chân 6) và dạng sóng trên tụ (chân 2).
- Nhận xét dạng sóng khi thay đổi VR.
- Tính tần số và biên độ của sóng ra khi RV1 ở vị trí Min, Max với tham số cho như trong hình.
- Căn chỉnh lại trị số của mạch để dạng sóng ra có tần số là 1Kz.



Gợi ý: Để có nguồn +5V và -5V bạn lấy nguồn ra sau đó thay đổi thành nguồn +5V và -5V theo yêu cầu.

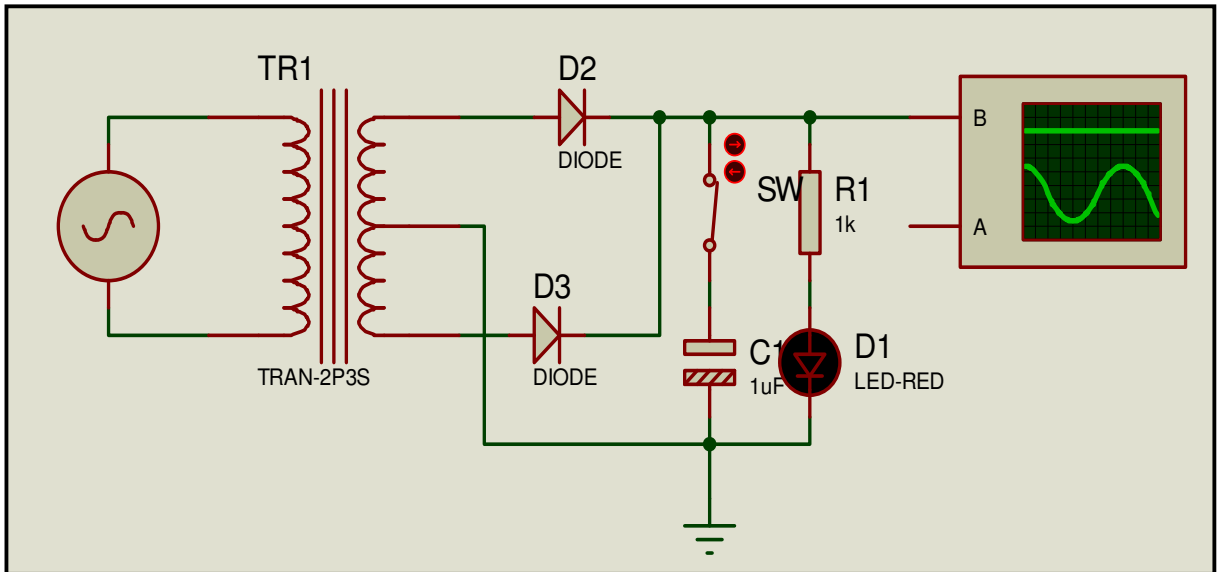


Bài 3:

- Vẽ và mô phỏng mạch chỉnh lưu như sau:
- Dùng máy hiện sóng đo dạng sóng tại đầu ra
- DC VOLMETER đo điện áp ngõ ra.
- DC AMMETER đo dòng điện ngõ ra.
- Nhận xét khi SW đóng và hở.
- Khi SW đóng thay đổi trị số tụ điện C1 và nhận xét dạng sóng đầu ra.

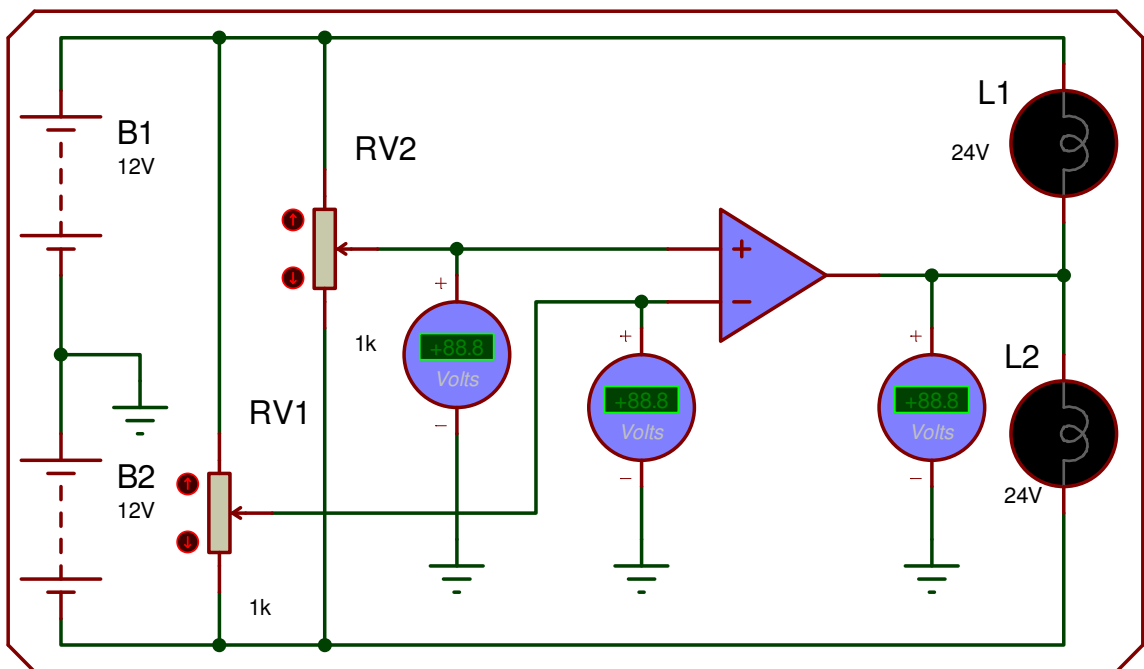


Gợi ý: Lấy DC VOLTMETER và DC AMMETER bạn vào sau đó chọn DC VOLTMETER và DC AMMETER , OSCILLOSCOPE theo yêu cầu:



Bài 4:

Khảo sát mạch điện sau:



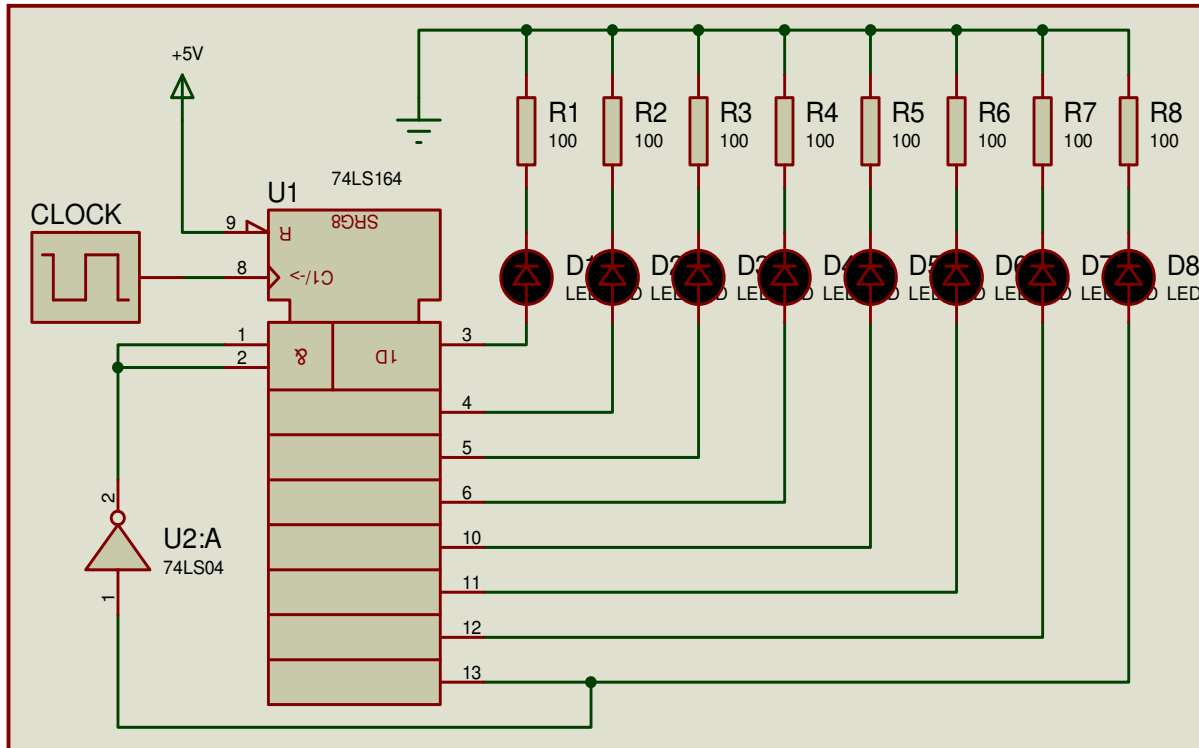
Thao tác: Vẽ lại mạch điện --> cho chạy mô phỏng --> điều chỉnh RV1 và RV2 --> quan sát trạng thái (sáng hay tắt) của các bóng đèn L1 và L2.

3.2. Thiết kế và mô phỏng mạch số

3.2.1. Thiết kế mạch quảng cáo.

Vẽ và mô phỏng mạch quảng cáo sáng dần dần như sau:

Nhận xét khi thay đổi tần số xung Clock.



Các bước tiến hành:

Bước 1: Lấy linh kiện

- Lấy IC 74LS164: Trong ô Keywords nhập 74LS164 --> trong ô DEVICE chọn 74LS164.IEC 74LS.
- Lấy IC 74LS04: Trong ô Keywords nhập 74LS04 --> trong ô DEVICE chọn 74LS04 74LS.
- Lấy xung CLOCK: Trong ô Keywords nhập Clock --> trong ô DEVICE chọn CLOCK ACTIVE.
- Lấy điện trở, LED RED, Nguồn cấp , GND làm tương tự như các bài trước.

Bước 2: Sắp xếp và đặt tên linh kiện

(Bạn thực hiện như phần bước 2 mục 3.1.1).

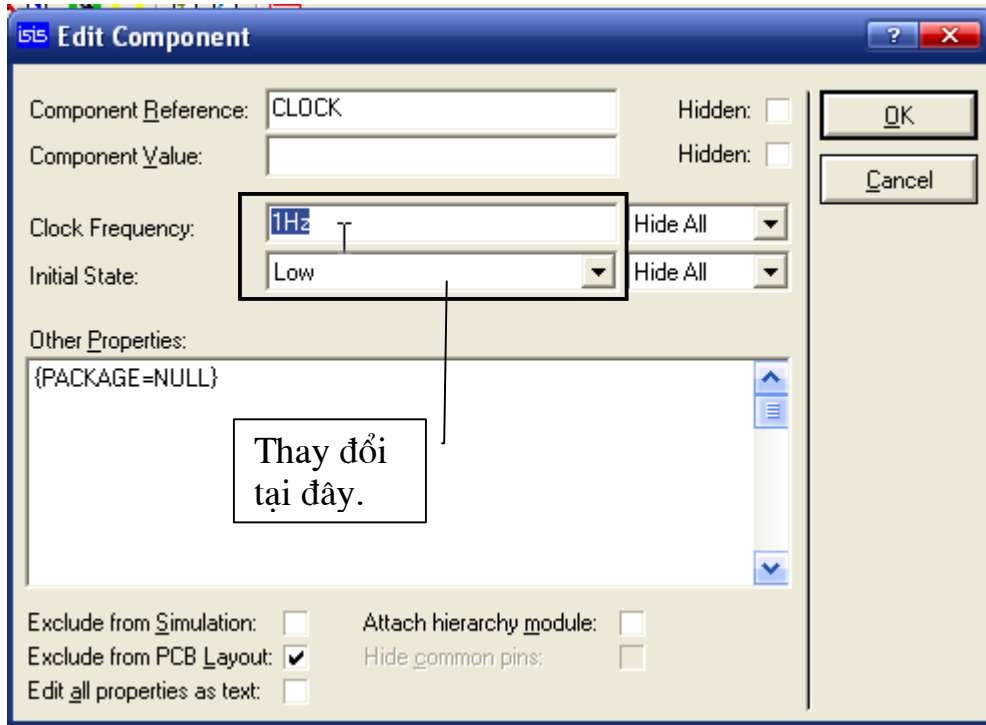
Bước 3: Kết nối mạch điện

(Bạn thực hiện như phần bước 3 mục 3.1.1).

Bước 3: Mô phỏng mạch điện:

Sau khi sắp xếp và đặt tên, thay đổi giá trị linh kiện như sơ đồ yêu cầu, bạn nhấn RUN để bắt đầu mô phỏng.

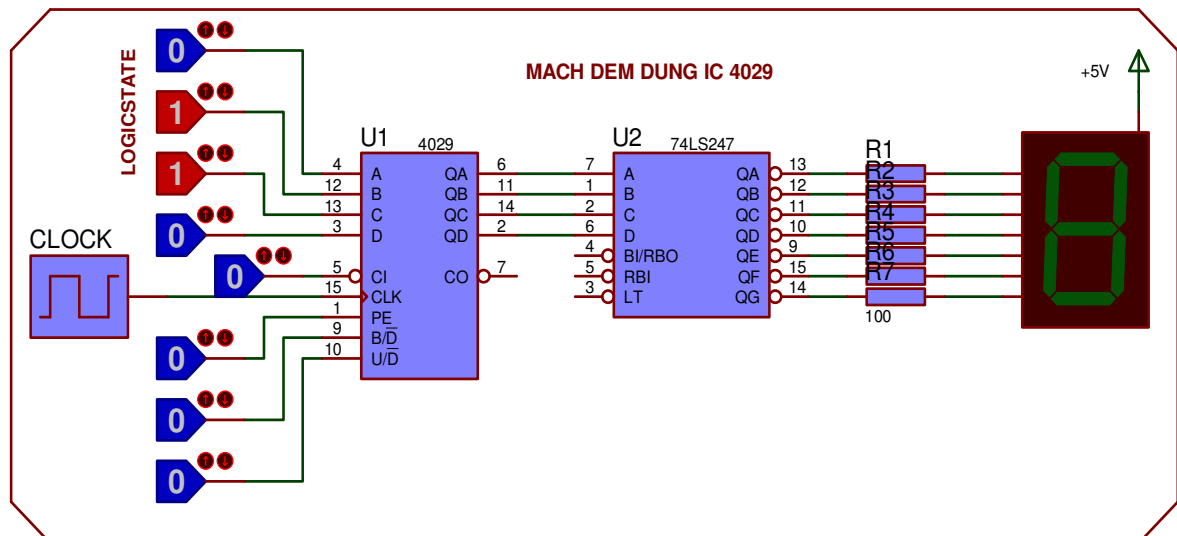
Các LED lần lượt sáng dần và tắt dần như ý tưởng chúng ta đưa ra:
 - Để thay đổi tần số xung CLOCK bạn nháy chuột phải vào đối tượng CLOCK (đối tượng chuyển sang màu đỏ) bạn nháy tiếp chuột trái cửa sổ Edit Component xuất hiện, bạn hãy quan sát hai ô Clock Frequency và Initial State hai ô này cho biết tần số xung đang sử dụng (1Hz) và Cảnh tác động của xung (cạnh xuống Low). Nếu cần thay đổi bạn thay đổi trong hai ô này.



3.2.2. Thiết kế mạch mô phỏng mạch đếm.

Vẽ và mô phỏng mạch đếm như sau:

Nhận xét khi thay đổi tần số xung Clock.



Một điểm mạnh của Proteus nữa là khi bạn thiết kế các mạch số; bằng các trạng thái logic sẵn có (logicstate) giúp bạn khảo sát nhanh hơn một mạch điện hay một IC số nào đó; việc tác động để làm thay đổi mức logic là rất đơn giản (chỉ cần nháy chuột); Logicstate chỉ có hai trạng thái duy nhất đó là mức 0 và mức 1 (mức 0 = 0V, mức 1= 5V). Bài tập này giúp bạn có được kỹ năng khi sử dụng các logicstate để khảo sát các bài toán.

Các bước tiến hành:

Bước 1: Lấy linh kiện

- Lấy IC 4029: Trong ô Keywords nhập 4029 --> trong ô DEVICE chọn 4029 CMOS.
- Lấy IC 74LS247: Trong ô Keywords nhập 74247 --> trong ô DEVICE chọn 74LS247 74LS.
- Lấy xung CLOCK: Trong ô Keywords nhập Clock --> trong ô DEVICE chọn CLOCK ACTIVE.
- Lấy Led 7 đoạn: Do IC giải mã ở đây dùng loại 74LS247 có ngõ ra tích cực mức thấp nên Led 7 đoạn ta chọn loại chung Anot. Vậy trong ô Keywords nhập 7seg --> trong ô DEVICE có rất nhiều lựa chọn cho bạn chọn, bạn chọn Led loại chung Anot với màu Led mà mình yêu thích (ở đây tôi chọn 7SEG-COM-AN-RGN DISPLAY.
- Lấy LOGICSTATE bạn vào Debugging Tools --> chọn LOGICSTATE.
- Lấy điện trở, nguồn cấp , GND làm tương tự như các bài trước.

Bước 2: Sắp xếp và đặt tên linh kiện.

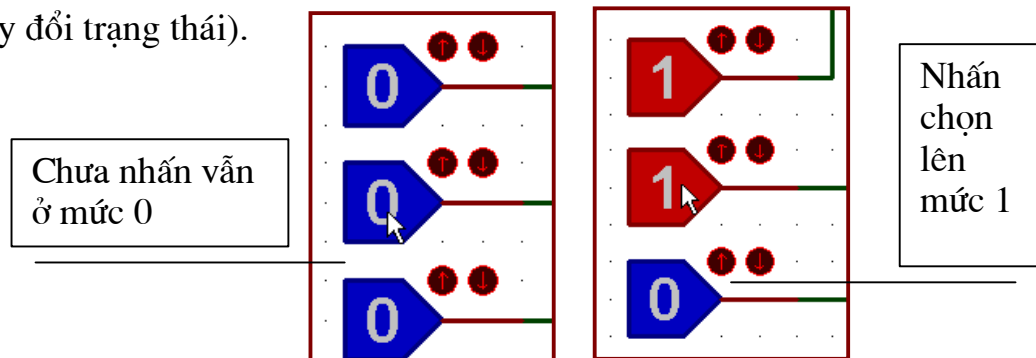
(Bạn thực hiện như phần bước 2 mục 3.1.1).

Bước 3: Kết nối mạch điện.

(Bạn thực hiện như phần bước 3 mục 3.1.1).

Bước 3: Mô phỏng mạch điện.

Sau khi sắp xếp và đặt tên, thay đổi giá trị linh kiện như sơ đồ yêu cầu, bạn nhấn RUN để bắt đầu mô phỏng. Bạn tiến hành khảo sát IC đếm 4029 bằng cách thay đổi các mức logic ở các đầu vào điều khiển (nháy chuột trực tiếp lên logicstate để làm thay đổi trạng thái).



3.2.3. Bài tập

Hãy vào Help \ Samples Designs \ Interactive Simulation Samples \ Animated Circuits: Khảo sát và vẽ mô phỏng lại các mạch trong các mục sau:

- Combinational Logic Circuit :
 - + Circuit 1: Khảo sát cổng AND.
 - + Circuit 2: Khảo sát cổng OR.
 - + Circuit 3: Khảo sát cổng NOT.
 - + Circuit 4: Khảo sát cổng NAND.
 - + Circuit 5: Khảo sát cổng NOR.
 - + Circuit 6: Khảo sát cổng XOR.
- Sequential Logic Circuit:
 - + Circuit 1: Khảo sát Flip – Flop RS.
 - + Circuit 2: Khảo sát Flip – Flop kiểu D.
 - + Circuit 3: Khảo sát Flip – Flop J-K .
 - + Circuit 4: Khảo sát mạch ghi dịch.
 - + Circuit 5: Khảo sát mạch đếm 3 bit nhị phân.
- Circuit For Fun
 - + Circuit 1: Khảo sát mạch đèn giao thông.
 - + Circuit 2: Khảo sát mạch đếm.

3.3. Thiết kế và mô phỏng chi vi điều khiển họ 8051

Vi điều khiển họ 8051 là một IC đang được sử dụng phổ biến nhất hiện nay bởi tính cơ động và giá thành cực rẻ của nó (chỉ khoảng 20.000VNĐ), việc lập trình (viết chương trình) có thể tiến hành trên nhiều phần mềm khác nhau như SIM51, Keil, Pinnacle 52, Bascom,... miễn làm sao biên dịch được sang đuôi “.Hex” hoặc đuôi “.Bin” để nạp vào bộ nhớ ROM của Vi điều khiển. Nhưng vấn đề khó khăn cho những người làm việc với vi điều khiển đó là về phần cứng; một chương trình muốn hoàn chỉnh thì phần cứng và phần mềm luôn phải đi liền với nhau, không những thế khi đã có phần cứng rồi để một chương trình viết ra nạp được vào bộ nhớ Rom của Vi điều khiển thì cần phải có sự hỗ trợ của các chương trình (thiết bị nạp chuyên dụng) thì mới thực hiện được. Hay nói cách khác để học về vi điều khiển người học cần đầu tư rất nhiều kinh phí. Để khắc phục các khó khăn trên bạn nên sử dụng phần mềm hỗ trợ mô phỏng Proteus để thiết kế

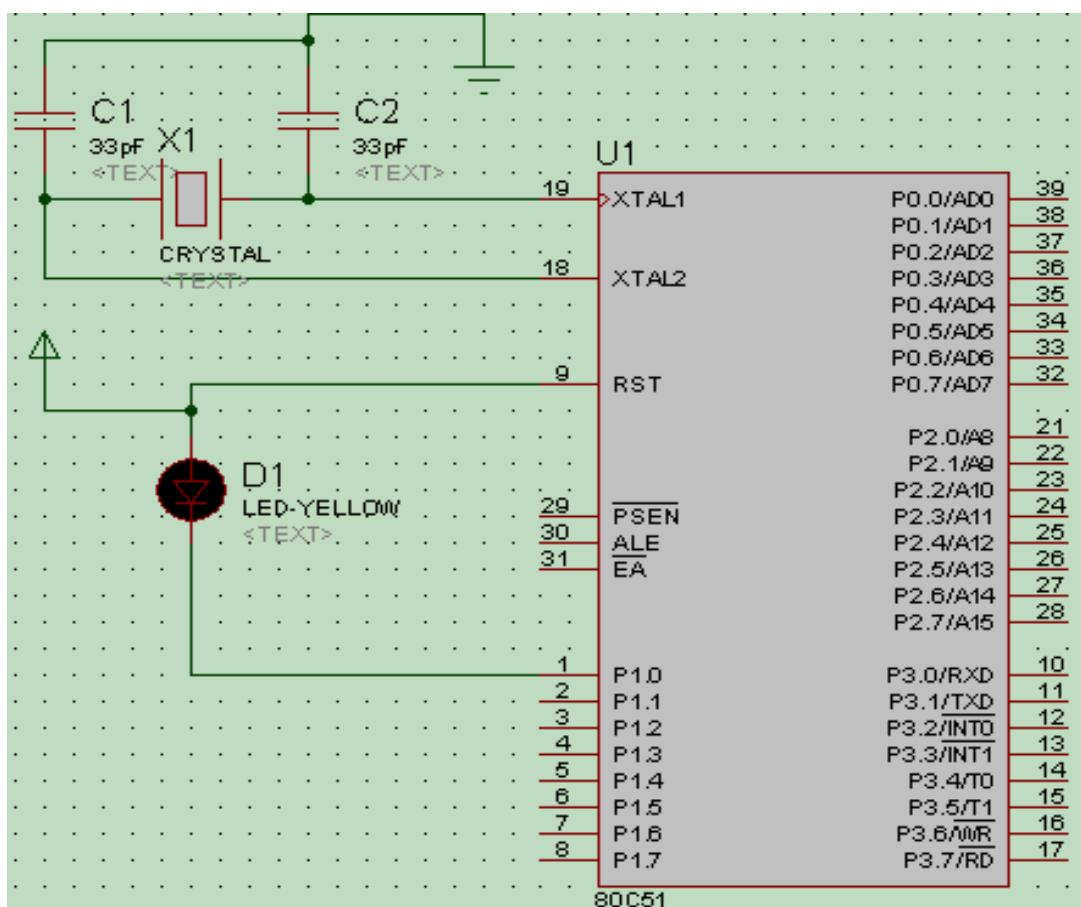
mô phỏng, với khả năng mô phỏng gần như với thực tế, việc cân chỉnh trị số cho phù hợp gần như sát với thực tế.

Một thế mạnh nữa của Vi điều khiển là tính mềm hóa chương trình tức là khi cần thay đổi yêu cầu hệ thống thì người dùng không cần phải thay đổi phần cứng mà chỉ cần thay đổi phần mềm việc này được thực hiện rất đơn giản trên máy tính./

Và thế mạnh của Proteus là chỉ với một mạch điện thể hiện phần cứng công nghệ bạn có thể nạp nhiều chương trình khác nhau để quan sát thử nghiệm (không mất nhiều công sức để thiết kế lại phần cứng).

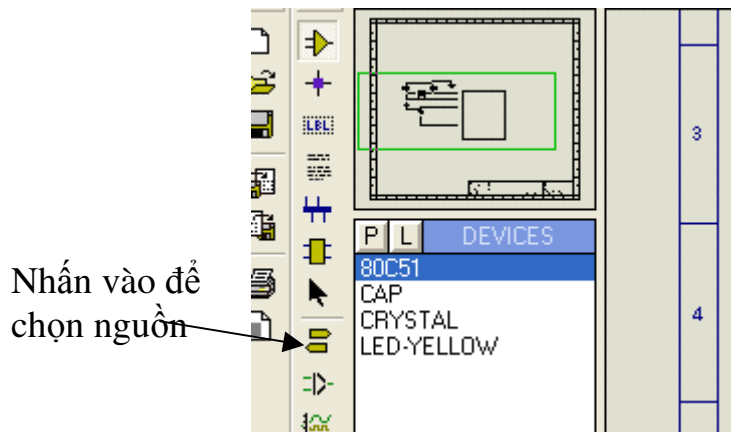
3.3.1. Mạch chớp tắt một Led đơn giản.

- **Thiết kế mạch nguyên lý:**



Bước 1. Chọn linh kiện:

- 8051: Vào Library/Pick (nhấn “p”): Catagory/Microprocessor ICs/80C51.
- Tụ: Nhấn “p”, trong Keywords đánh Capacitor; trong Sub-Catagory\ Generic \ Cap rồi đổi giá trị của tụ thành 33pF.
- LED: Nhấn “p”/ Keywords=Led, Catagory/Optoelectronic/Led-Yellow.
- Keywords=Crystal, để lấy thạch anh, thay đổi giá trị =12MHz.
- Nguồn cấp: 5V(Vcc) và 0V(GND):



Chọn nguồn Vcc=Power, GND=Ground.

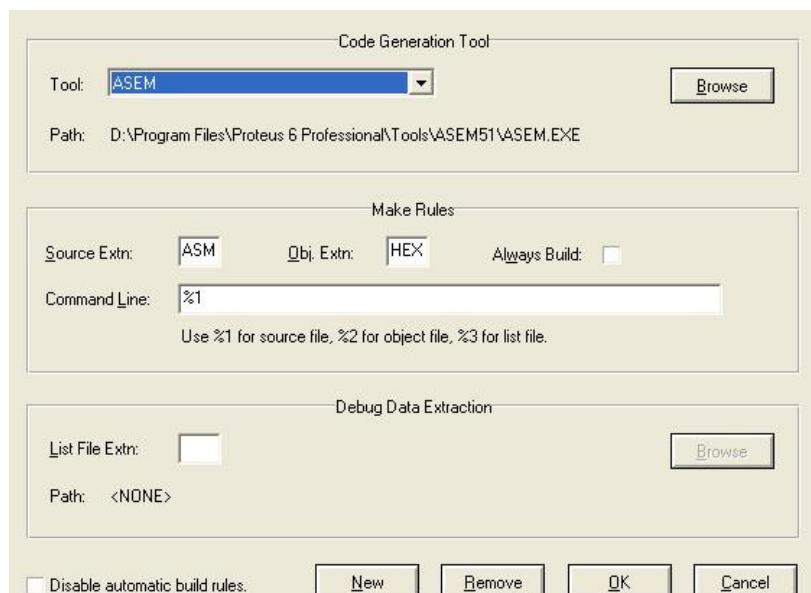
Bước 2. Đi dây.

Chọn Tools/Wire (nhấn “w”). Di chuyển chuột đến chân linh kiện hay đường dây, sẽ hiện 2 đường chéo. Nhấn chuột trái để nối dây.

Bước 3. Mô phỏng.

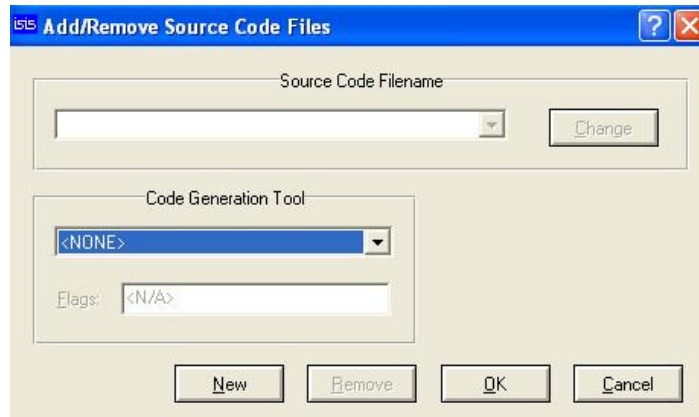
Các bạn cũng biết, để 8051 hoạt động, trước hết cần viết chương trình (bằng Asm hay C), dùng phần mềm biên dịch chuyển sang .Hex, cuối cùng nạp vào IC. Với Proteus, hoàn toàn tương tự.

Đầu tiên, hãy xác định phần mềm biên dịch (Code Generator), chọn: Source/Define Code Generation Tools, hiện :



Chọn Browse để link đến chương trình biên dịch Proteus đã cài sẵn.

Sau đó, viết chương trình, nhấn Source/ Add Source file, hiện cửa sổ:



Chọn Code Generation Tool là ASEM, chọn New để tạo file mới, đánh tên file bất kỳ*.asm. (VD: NHAY.asm.) Copy đoạn mã viết bằng ASM sau:

ORG 0000H	Delay:
Start:	MOV R0,#0FFH
CPL P1.0	LOOP:MOV R1,#0FFH
ACALL Delay	DJNZ R1,\$
SJMP start	DJNZ R0,LOOP
	RET
	END

Tiếp theo, biên dịch để tạo file .Hex, nhấn Source/Build All. Hiện thông báo “Built Complete OK”, nếu không trình biên dịch sẽ báo lỗi, kèm theo vị trí lỗi.

Cuối cùng, nạp file .Hex vào IC bằng cách vào bảng Edit của 80C51, trong Program File/link đến file .Hex vừa tạo. Đừng quên thay đổi Clock Frequency=12MHz. Nhấn nút chạy chương trình nằm ở phía dưới màn hình.



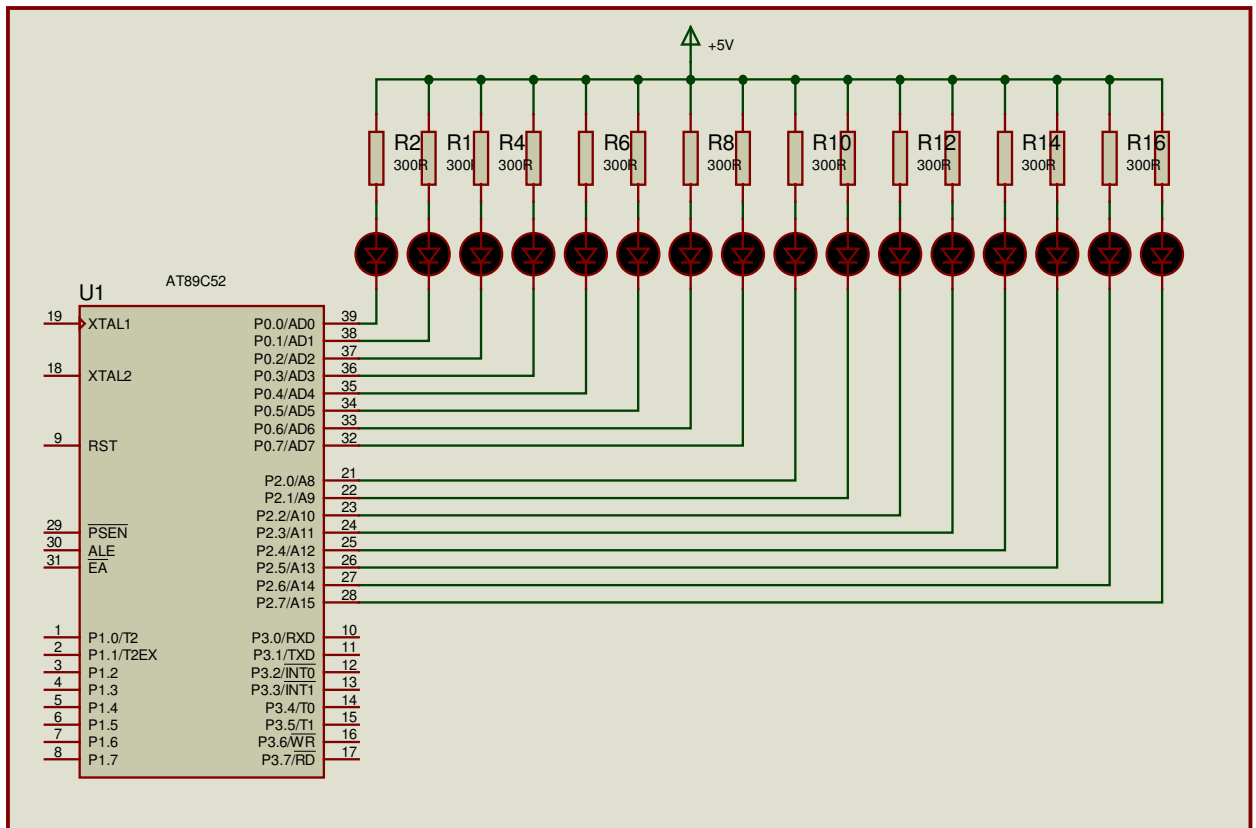
Ngoài ra bạn cũng có thể soạn thảo nội dung chương trình trên bất kì phần mềm soạn thảo nào, sau đó thực hiện biên dịch để tạo ra được File .HEX. Tiếp đó bạn hãy COPY File . HEX vừa tạo tới thư mục chứa File sơ đồ nguyên lý của Proteus. Rồi tiến hành nạp vào bộ nhớ của Vi điều khiển như trên.

3.3.2. Thiết kế và mô phỏng mạch quảng cáo hiện thị trên led đơn (16 led)

a. Thiết kế phần cứng (mạch nguyên lý)

Bạn hãy thiết kế sơ đồ nguyên lý như hình dưới đây:

Mạch sử dụng hai Port để hiển thị các Led, trị số điện trở hạn dòng cho Led qua tính toán trên lý thuyết ta chọn là 330R.



Các bước thực hiện:

Bước 1. Chọn linh kiện:

- AT89C52: Vào Library/Pick (nhấn “p”): Catagory/Microprocessor ICs \ AT89C52.
- Điện trở R: Nhấn “p”/ Keywords= R esistors; trong Sub-Catagory\ Generic \ RES, rồi đổi giá trị thành 330.
- LED: Nhấn “p”/ Keywords=Led, Catagory/Optoelectronic/LED-RED.
- Nguồn cấp: 5V(Vcc) và 0V(GND):

Bước 2. Đi dây.

Chọn Tools/Wire (nhấn “w”). Di chuyển chuột đến chân linh kiện hay đường dây, sẽ hiện 2 đường chéo. Nhấn chuột trái để nối dây.

Bước 3. Lưu File

Bạn lưu trên đĩa cứng của mình. Giả sử của Tôi lưu như sau: D\BT_PROTEUS\QUANGCAO.

b. Viết chương trình phần mềm.

Chương trình phần mềm trong bài này được biên soạn trên phần mềm soạn thảo Pinnacle 52 , mã nguồn lưu ở dạng đuôi “.ASM” (ví dụ LED0.ASM).

Mã nguồn file.asm

```

;-----
org 0000h          ;khai báo địa chỉ bắt đầu của chương trình (0000h).
main:              ; tên nhãn là main
    call choptat   ; gọi chương trình con chớp tắt
    call tatdan    ; gọi chương trình con tắt dần
    call toichay   ; gọi chương trình con một điểm tối chạy
    call ngoaivao  ; gọi chương trình con tắt dần từ giữa ra
    sjmp main      ; lặp lại từ đầu
;-----Đoạn chương trình con chớp tắt-----
choptat:
    mov r0,#10    ; biến đếm thực hiện 10 lần
choptat1:
    mov p0,#00h   ;tắt p0
    mov p2,#00h   ;tắt p2
    lcall delay   ;gọi chương trình con tạo tre
    mov p0,#0ffh  ;sang p0
    mov p2,#0ffh  ;sang p2
    lcall delay
    djnz r0,choptat1 ; nếu R0 chưa =10 thì quay về thực hiện tiếp
    ret           ; kết thúc chương trình con
;----- Đoạn chương trình con tat dan -----
tatdan:
    mov r0,#05h   ; chuyển 05h vào thanh ghi R5
tatdan0:
    mov p0,#00h   ; Xóa cổng P0
    mov p2,#00h   ; Xóa cổng P2

```

```
call delay          ; Gọi chương trình con tạo trễ
clr a               ; Xóa thanh ghi A
tatdan1:
setb c              ; Đặt cờ C lên 1
rlc a               ; Xoay trái cờ nội dung cờ C qua thanh ghi A
mov p0,a            ; Đưa ra P0 để hiển thị
call delay
jnc tatdan1        ; nhảy tới nhãn 'tatdan1' khi C=0
clr a               ; Xóa thanh ghi A
tatdan2:
setb c              ; Đặt cờ C lên 1
rlc a               ; Xoay trái cờ nội dung cờ C qua thanh ghi A
mov p2,a            ; Đưa ra P2 để hiển thị
call delay
jnc tatdan2        ; nhảy tới nhãn 'tatdan2' khi C=0
djnz r0,tatdan0    ; nhảy tới nhãn 'tatdan0' khi R0 chưa bằng 0
ret                 ; Kết thúc chương trình con tatdan
;-----Đoạn chương trình con một điểm tôi chạy-----
toichay:
mov r0,#05h        ; Biến đếm thực hiện 5 lần
toichay0:
mov p0,#00h        ; Xóa cổng P0
mov p2,#00h        ; Xóa cổng P1
call delay
clr a               ; Xóa thanh ghi A
setb c              ; đặt cờ C lên 1
toichay1:
rlc a               ; Xoay trái cờ C qua thanh ghi A
mov p0,a            ; nội dung thanh ghi A đưa ra hiển thị
call delay
jnc toichay1       ; nhảy tới nhãn 'toichay1' khi C=0
clr a               ; Xóa thanh ghi A
```



```
setb c                ; đặt cờ C lên 1
toichay2:
rlc a                 ; Xóa thanh ghi A
mov p2,a              ; nội dung thanh ghi A đưa ra hiển thị
call delay
jnc toichay2          ; nhảy tới nhãn 'toichay2' khi C=0
djnz r0,toichay0      ; nhảy tới nhãn 'toichay0' khi R0 chưa bằng 0
ret                   ; Kết thúc chương trình con toichay
;-----Đoạn chương trình con tắt dần từ giữa ra 2 bên -----
giuara:
mov r0,#10h           ; Biến đếm thực hiện 10 lần
giuara1:
mov p0,#00h           ; Xóa cổng P0
mov p2,#00h           ; Xóa cổng P2
call delay
mov p0,#80h           ; 08h đưa ra P0 hiển thị
mov p2,#01h           ; 01h đưa ra P2 hiển thị
call delay
mov p0,#0c0h          ; 0c0h đưa ra P0 hiển thị
mov p2,#03h           ; 03h đưa ra P2 hiển thị
call delay
mov p0,#0e0h          ; 0e0h đưa ra P0 hiển thị
mov p2,#07h           ; 07h đưa ra P2 hiển thị
call delay
mov p0,#0f0h          ; 0f0h đưa ra P0 hiển thị
mov p2,#0fh           ; 0fh đưa ra P2 hiển thị
call delay
mov p0,#0f8h          ; 0f8h đưa ra P0 hiển thị
mov p2,#1fh           ; 1fh đưa ra P2 hiển thị
call delay
mov p0,#0fch          ; 0fch đưa ra P0 hiển thị
mov p1,#3fh           ; 3fh đưa ra P2 hiển thị
```

```

call delay
mov p0,#0feh          ; 0feh đưa ra P0 hiển thị
mov p2,#7fh           ; 7fh đưa ra P2 hiển thị
call delay
mov p0,#0ffh          ; 0ffh đưa ra P0 hiển thị
mov p2,#0ffh          ; 0ffh đưa ra P2 hiển thị
call delay
djnz r0, giuara1      ; nhảy tới giuara1 khi R0 chưa bằng 0
ret
;-----Đoạn chương trình con delay -----
delay:
    mov r6,#0ffh      ;chuyển FFh vào R6
dl1:  mov r7,#0ffh      ;chuyển FFh vào R6
    djnz r7,$          ; giảm và nhảy tại chỗ khi R7 chưa bằng 0
    djnz r6,dl1        ; giảm và nhảy tới 'dl1' khi R6 chưa bằng 0
    ret                ; Kết thúc chương trình con tạo trễ
    end                ; Kết thúc chương trình chính.
; -----

```

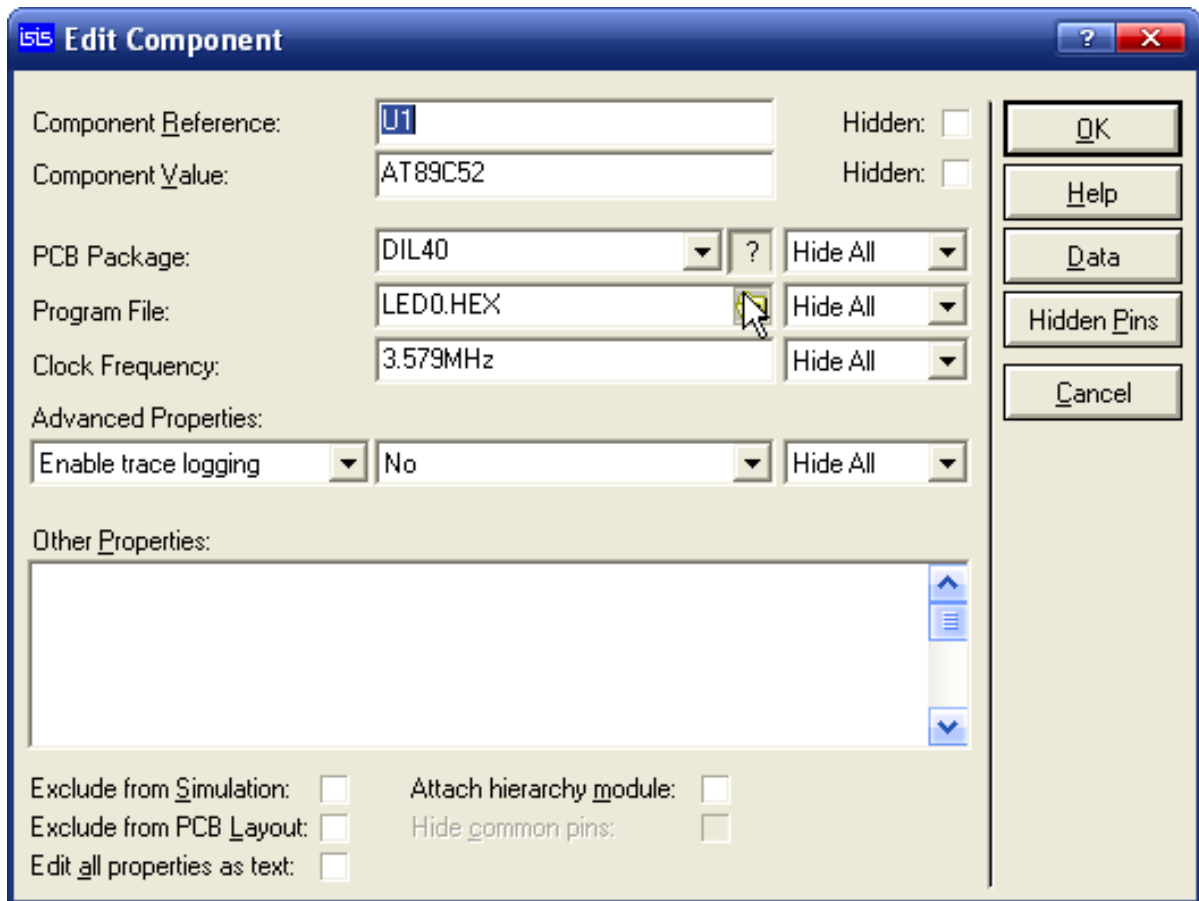
Sau khi soạn thảo xong mã nguồn của chương trình bạn thực hiện biên dịch sang đuôi .HEX; nếu trước đó File nguồn của bạn không lưu ở ổ D\BT_PROTEUS\QUANGCAO thì bạn hãy COPY File .HEX (LED0.HEX) vừa tạo được vào trong thư mục BT_PROTEUS như đường dẫn trên. (Còn nếu không thì cho dù bạn có nạp được chương trình cho IC thì mạch cũng không chạy).

c. Mô phỏng chương trình.

Quá trình mô phỏng gồm các công việc sau:

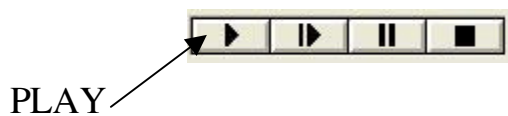
- Nạp chương trình lên IC.
- Thiết lập các thông số hoạt động.
- RUN rồi quan sát mô phỏng.

Đầu tiên, nháy phải chuột vào IC AT89C52 (IC chuyển màu đỏ) --> Nháy chuột trái lên IC --> cửa sổ hiện ra như sau:



- Trong ô Program File nháy chuột vào biểu tượng OPEN sao đó chọn đường dẫn tới thư mục chứa File LED0.HEX rồi nháy chọn File này; Trong ô Clock Frequency bạn hãy thay đổi lấy một tần số thích hợp để dễ quan sát--> cuối cùng nháy chọn OK. Như vậy là bạn đã thực hiện nạp xong chương trình lên IC với tần số lựa chọn là 3.579MHz.

- Nháy PLAY để chạy chương trình.



- Quan sát hoạt động của chương trình.

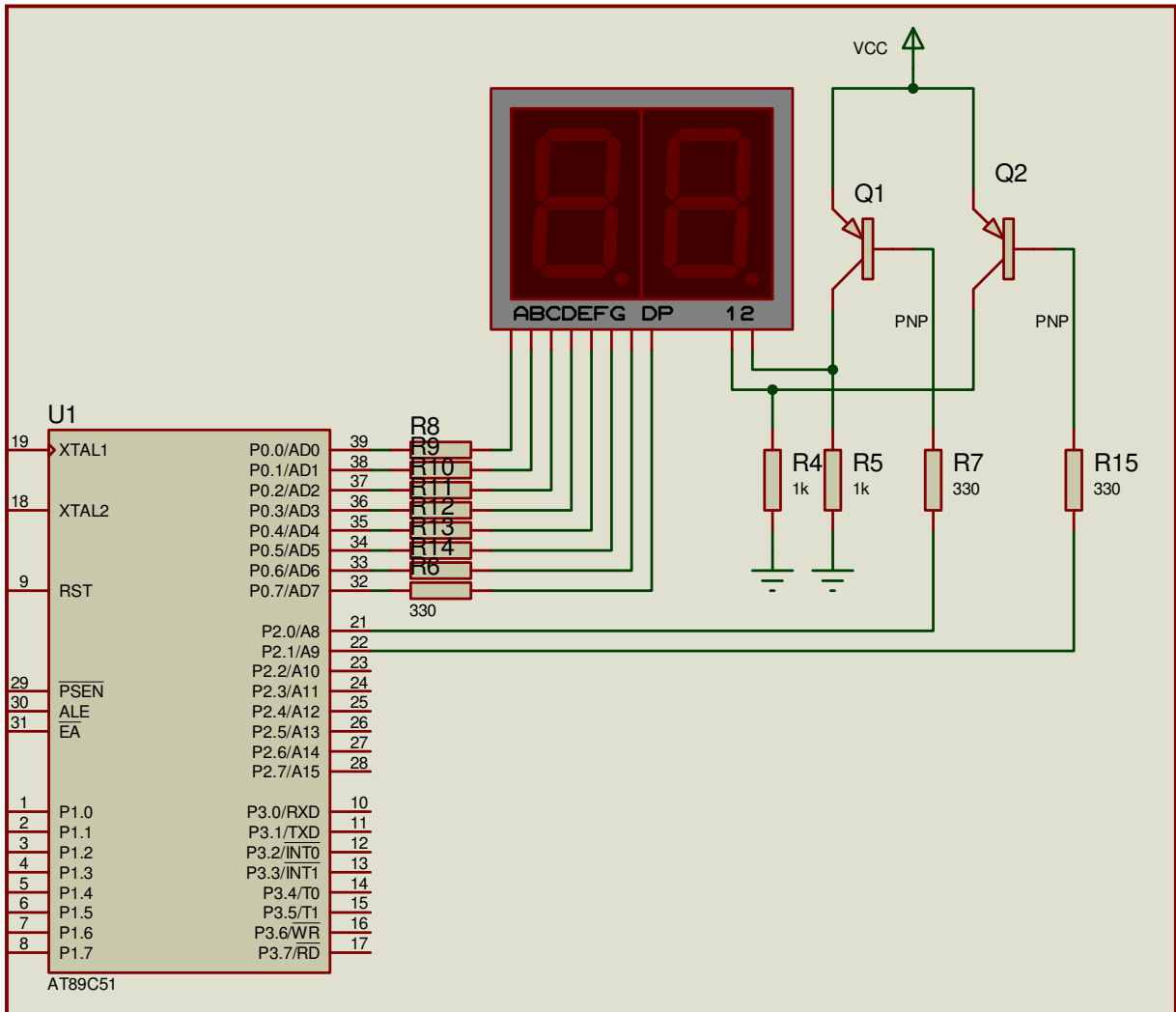
- Nhận xét khi thay đổi tần số trong mục Clock Frequency.

3.3.3. Bài tập:

Bài 1: Vẽ và mô phỏng mạch đếm lên 00 --> 99 hiển thị trên 2 led 7 đoạn:

(Sử dụng phương pháp quét Led – 2 Led).

a. Sơ đồ phân cứng:



b. Mã nguồn chương trình:

ORG 0000H

MAIN:

MOV R0,#0 ; biến đếm bắt đầu từ không (0).

MOV DPTR,#BANG_MA ; Nạp địa chỉ bảng mã vào DPTR

LAP:

MOV R1,#200 ; lặp lại 200 lần để quan sát

LAP2:

```

CALL GIAIMA                ; gọi chương trình con giải mã
CALL HIEN_THI              ; gọi chương trình con hiển thị
DJNZ R1,LAP2
INC R0                      ; tăng R0 lên 1 đơn vị
CJNE R0,#100,LAP          ; So sánh với 100
JMP MAIN

```

;----- Chương trình con giải mã -----

GIAIMA:

```

MOV A,R0                   ; chuyển R0 vào thanh ghi A
MOV B,#10                  ; Nạp giá trị 10 vào thanh ghi B
DIV AB                     ; Chia A cho B
MOVC A,@A+DPTR             ; A chứa giá trị hàng chục
MOV 20H,A                  ; giá trị hàng chục lưu tại ô nhớ 20H
MOV A,B
MOVC A,@A+DPTR            ; A chứa giá trị hàng đơn vị
MOV 21H,A                  ; giá trị hàng đơn vị lưu tại ô nhớ 21H
RET

```

;----- chương trình con hiển thị -----

HIEN_THI:

```

MOV P0,21H                ; Giá trị đơn vị đưa ra hiển thị
MOV P2,#1111110B          ; Cho Led đơn vị sáng
CALL DELAY
MOV P2,#0FFH              ; tắt hết để chống lem

MOV P0,20H                ; Giá trị hàng chục đưa ra hiển thị
MOV P2,#11111101B         ; Cho Led chục sáng
CALL DELAY
MOV P2,#0FFH              ; tắt hết để chống lem
RET

```

;----- chương trình con tạo thời gian trễ để quét Led -----

DELAY:

```

MOV R7,#255
DJNZ R7,$
RET

```

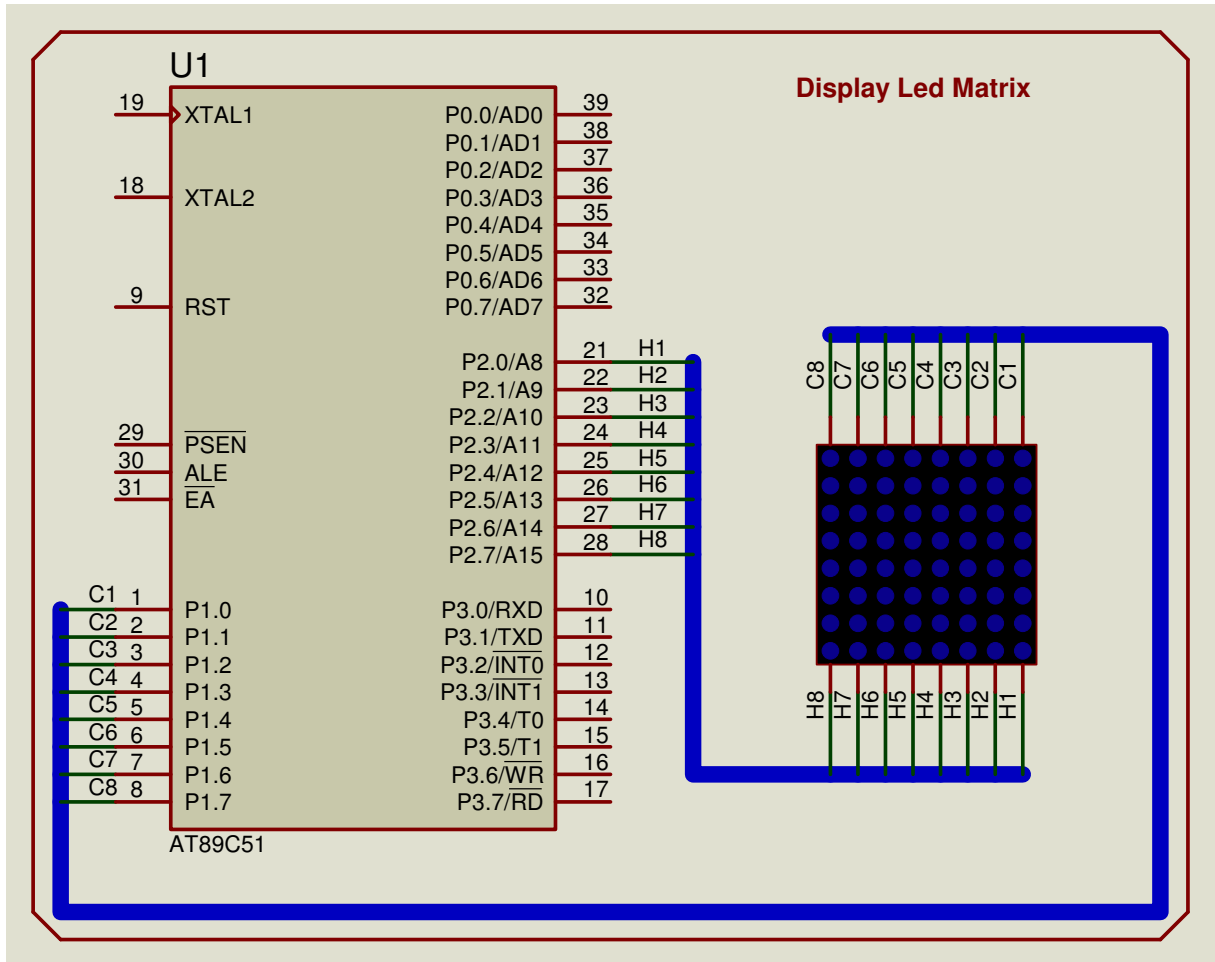
;----- Bảng mã số 7 đoạn -----

BANG_MA:

DB 0C0H,0F9H,0A4H,0B0H,99H,92H,82H,0F8H,80H,90H

END

Bài 2: Vẽ và mô phỏng mạch hiển thị ma trận Led:



Gợi ý: Bạn nháy vào nút công cụ BUS để vẽ đường Bus



Mã nguồn chương trình:

; chương trình hiển thị chữ H trên Led ma trận

ORG 00H

Main:

MOV A,#11001100B

MOV B,#00000000B

CALL MTX

MOV A,#11111100B

MOV B,#11100111B

CALL MTX

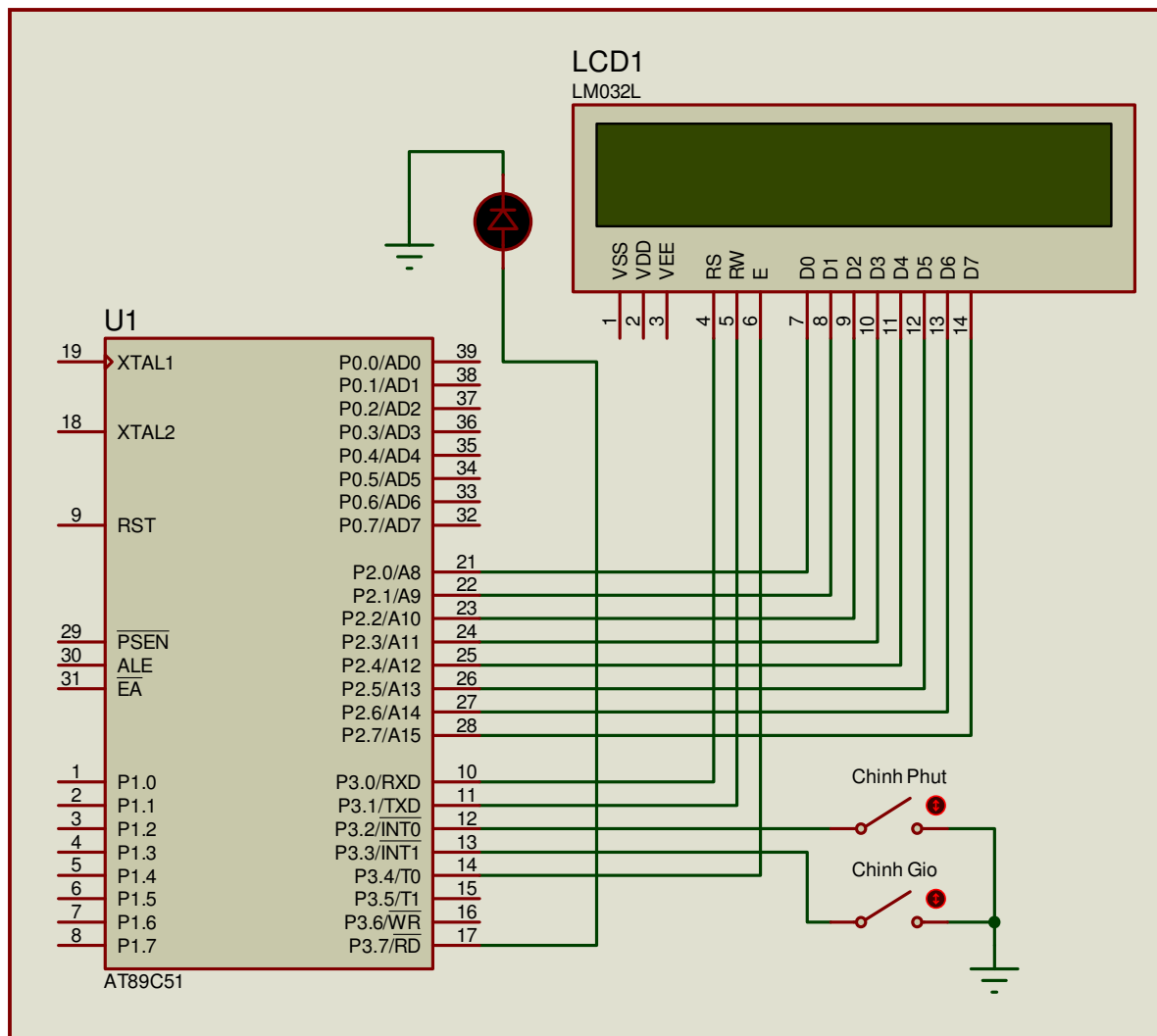
LJMP Main

```

MTX:
    MOV R1,#1
LOOP1:
    MOV R2,#5
LOOP2:
    MOV R3,#255
LOOP3:
    MOV P1,A
    MOV P1,#00000000B
    MOV P2,B
    DJNZ R3,LOOP3
    DJNZ R2,LOOP2
    DJNZ R1,LOOP1
    RET
    END
    
```

Bài 3: Vẽ và mô phỏng mạch đồng hồ điện tử hiển thị trên LCD như sau:

a. Sơ đồ phân cứng:



b. Mã nguồn ASM

```

enable_lcd equ p3.4
rs equ p3.0
rw equ p3.1
lcd_data equ p2
;=====
org 0000h
ljmp main
org 0003h
ljmp int0isr
org 0013h
ljmp int1isr
org 000bh
ljmp timer0isr
org 0040h
main:
mov sp,#5fh
lcall init_system
main_loop:
lcall test1s
sjmp main_loop
;-----
init_system:
mov 30h,#'0'
mov 31h,#'0'
mov 32h,#'0'
mov 33h,#'0'
mov 34h,#'0'
mov 35h,#'0'
mov tmod,#01h
mov th0,#high(-10000)
mov tl0,#low(-10000)
setb ex0
setb ex1
setb it0
setb it1
setb et0
setb ea
setb tr0
mov a,#01h
lcall write_cmd
mov a,#0ch
lcall write_cmd
mov a,#38h
lcall write_cmd
ret
;-----
int0isr:
push acc
inc 33h
mov a,33h
cjne a,#':',exit_int0
mov 33h,#'0'
inc 32h
mov a,32h

```



```

    cjne a,#'6',exit_int0
    mov  32h,#'0'

    inc  31h
    mov  a,30h
    cjne a,#'1',next_int0
    mov  a,31h
    cjne a,#'3',exit_int0
    mov  31h,#'1'
    mov  30h,#'0'
next_int0:
    mov  a,31h
    cjne a,#':',exit_int0
    mov  31h,#'0'
    inc  30h
exit_int0:
    mov  34h,#'0'
    mov  35h,#'0'
    pop  acc
    reti
;-----
int1isr:
    push acc
    inc  31h
    mov  a,30h
    cjne a,#'1',next_int1
    mov  a,31h
    cjne a,#'3',exit_int1
    mov  31h,#'1'
    mov  30h,#'0'
next_int1:
    mov  a,31h
    cjne a,#':',exit_int1
    mov  31h,#'0'
    inc  30h

```

```

exit_int1:
    mov  34h,#'0'
    mov  35h,#'0'
    pop  acc
    reti
;-----
timer0isr:
    push acc
    inc  r1
    mov  th0,#high(-10000)
    mov  tl0,#low(-10000)
    pop  acc
    reti
write_cmd:
    clr  rs
    clr  rw
    mov  lcd_data,a
    nop
    setb enable_lcd
    nop
    clr  enable_lcd
    lcall delay1ms
    ret
;-----
write_data:
    setb rs
    clr  rw
    mov  lcd_data,a
    nop
    setb enable_lcd
    nop
    clr  enable_lcd
    lcall delay1ms
    ret
;-----

```

display:

```

mov  a,#80h
lcall write_cmd
mov  a,30h
lcall write_data
mov  a,31h
lcall write_data
mov  a,#83h
lcall write_cmd
mov  a,32h
lcall write_data
mov  a,33h
lcall write_data
mov  a,#86h
lcall write_cmd
mov  a,34h
lcall write_data
mov  a,35h
lcall write_data
ret

```

;-----

test1s:

```

cjne r1,#100,exit
lcall display
cpl  p3.7
mov  r1,#00h
inc  35h
mov  a,35h
cjne a,#':',exit
mov  35h,#'0'
inc  34h
mov  a,34h
cjne a,#'6',exit
mov  34h,#'0'
inc  33h

```

```

mov  a,33h
cjne a,#':',exit
mov  33h,#'0'
inc  32h
mov  a,32h
cjne a,#'6',exit
mov  32h,#'0'
inc  31h
mov  a,30h
cjne a,#'1',next
mov  a,31h
cjne a,#'3',exit
mov  31h,#'1'
mov  30h,#'0'

```

next:

```

mov  a,31h
cjne a,#':',exit
mov  31h,#'0'
inc  30h

```

exit:

ret

;-----

delay1ms:

```

mov  r1,#50
delay1ms_loop:
mov  r2,#25
djnz r2,$
djnz r1,delay1ms_loop
ret
end

```

MỤC LỤC

	Trang
GIỚI THIỆU PHẦN MỀM	1
CHƯƠNG 1: HƯỚNG DẪN CÀI ĐẶT PHẦN MỀM	2
1.1. Yêu cầu cấu hình.....	2
1.2. Các bước cài đặt phần mềm lên đĩa cứng.....	2
1.3. Khởi động và thoát khỏi chương trình.....	8
a. Khởi động chương trình.....	8
b. Thoát khỏi chương trình.....	10
CHƯƠNG 2: GIỚI THIỆU GIAO DIỆN PHẦN MỀM	12
2.1. Giới thiệu chung.....	12
2.2. Thứ tự xếp đặt trên màn hình (Screen layout).....	12
2.3. Các thao tác cơ bản hay sử dụng.....	16
CHƯƠNG 3: THỰC HÀNH THIẾT KẾ VÀ MÔ	17
PHỎNG MẠCH ĐIỆN	
3.1. Thiết kế và mô phỏng mạch tương tự.....	17
3.1.1. Mô phỏng hoạt động của Diode.....	17
3.1.2. Mạch tạo dao động dùng IC 555.....	22
3.1.3. Bài tập	28
3.2. Thiết kế và mô phỏng mạch số.....	31
3.2.1. Thiết kế mạch quảng cáo.....	31
3.2.2. Thiết kế mạch mô phỏng mạch đếm.....	32
3.2.3. Bài tập	34
3.3. Thiết kế và mô phỏng chi vi điều khiển họ 8051	34
3.3.1. Mạch chớp tắt một Led đơn giản.....	35
3.3.2. Thiết kế và mô phỏng mạch quảng cáo hiện thị trên led đơn (16 led)	38
a. Thiết kế phần cứng (mạch nguyên lý)	38
b. Viết chương trình phần mềm.....	39
c. Mô phỏng chương trình.....	42
3.3.3. Bài tập	44
4.	

