

Chương III

XÂY DỰNG PHẦN MỀM

Biên soạn : Lâm tăng Đức – Lê Tiến Dũng – Bộ môn TĐH

cuu duong than cong . com

cuu duong than cong . com

Chương III

XÂY DỰNG PHẦN MỀM

3.1 Chương trình MONITOR cho KIT

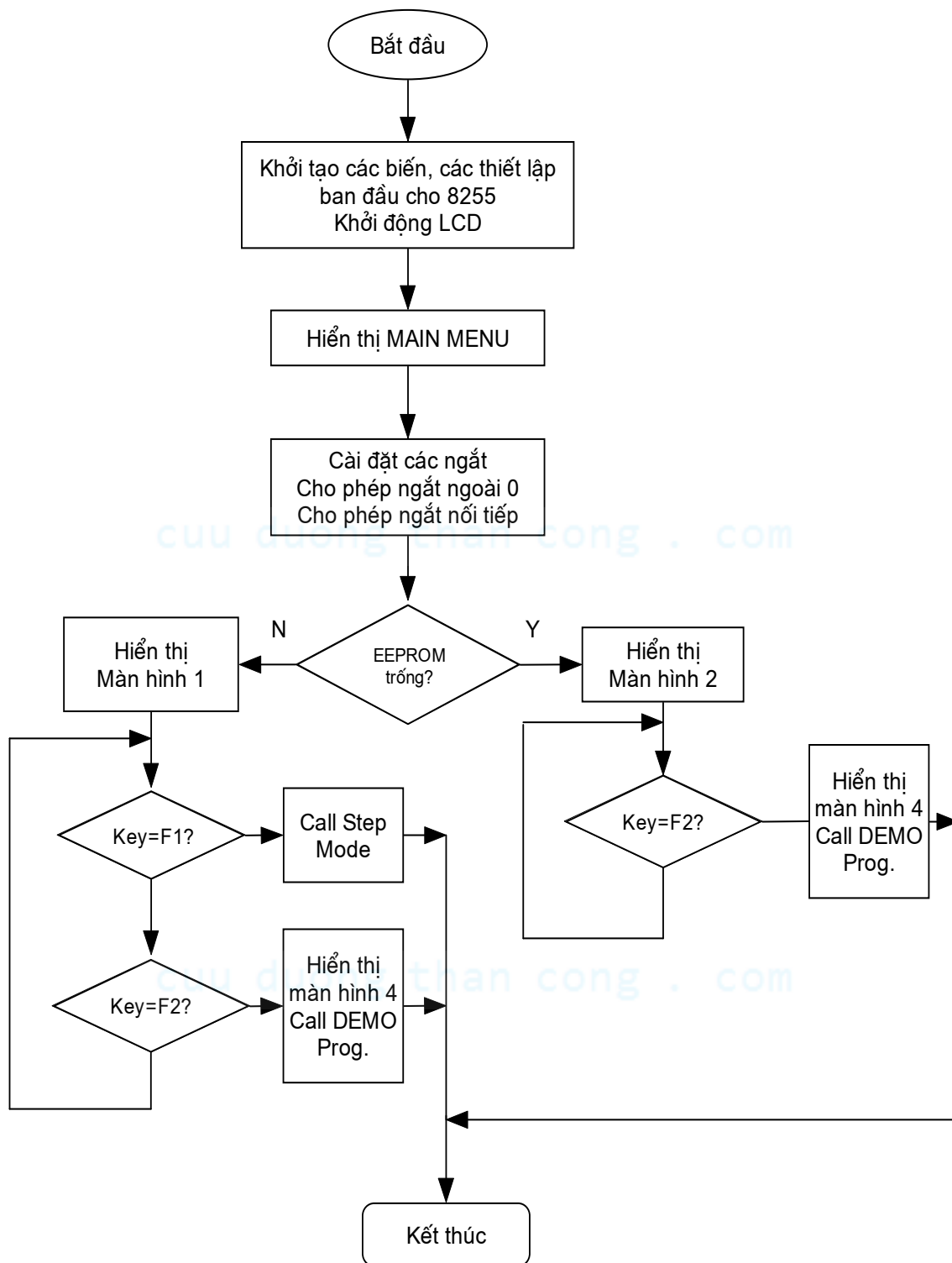
Do chương trình MONITOR của hệ thống phức tạp và khá lớn nên ta sẽ tổ chức thành nhiều module nhỏ ghép lại với nhau. Mỗi module thực hiện mỗi chức năng riêng, các module trong chương trình:

- + Module thực hiện việc hiển thị LCD.
- + Module quản lý bàn phím.
- + Module quản lý việc đọc/ghi dữ liệu trong EEPROM và RAM.
- + Module tạo thời gian trễ.
- + Chương trình chính (Main).
- + Module điều khiển truy cập các thanh ghi đặc biệt và các ô nhớ bên trong chip vi điều khiển chính.
- + Module truyền thông giao tiếp với máy tính PC.

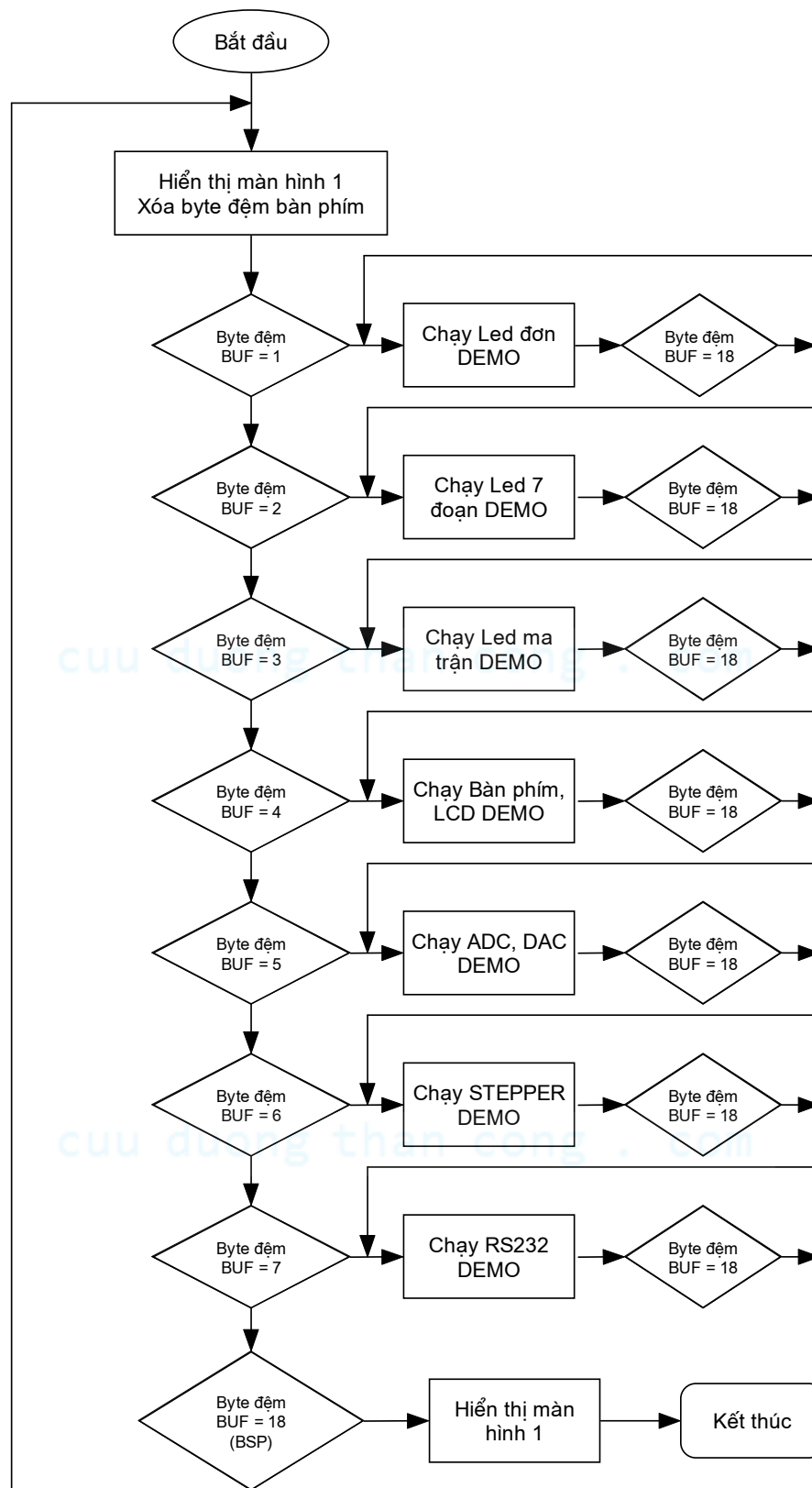
Mã nguồn của chương trình hệ điều hành được giới thiệu ở phần phụ lục của đồ án.

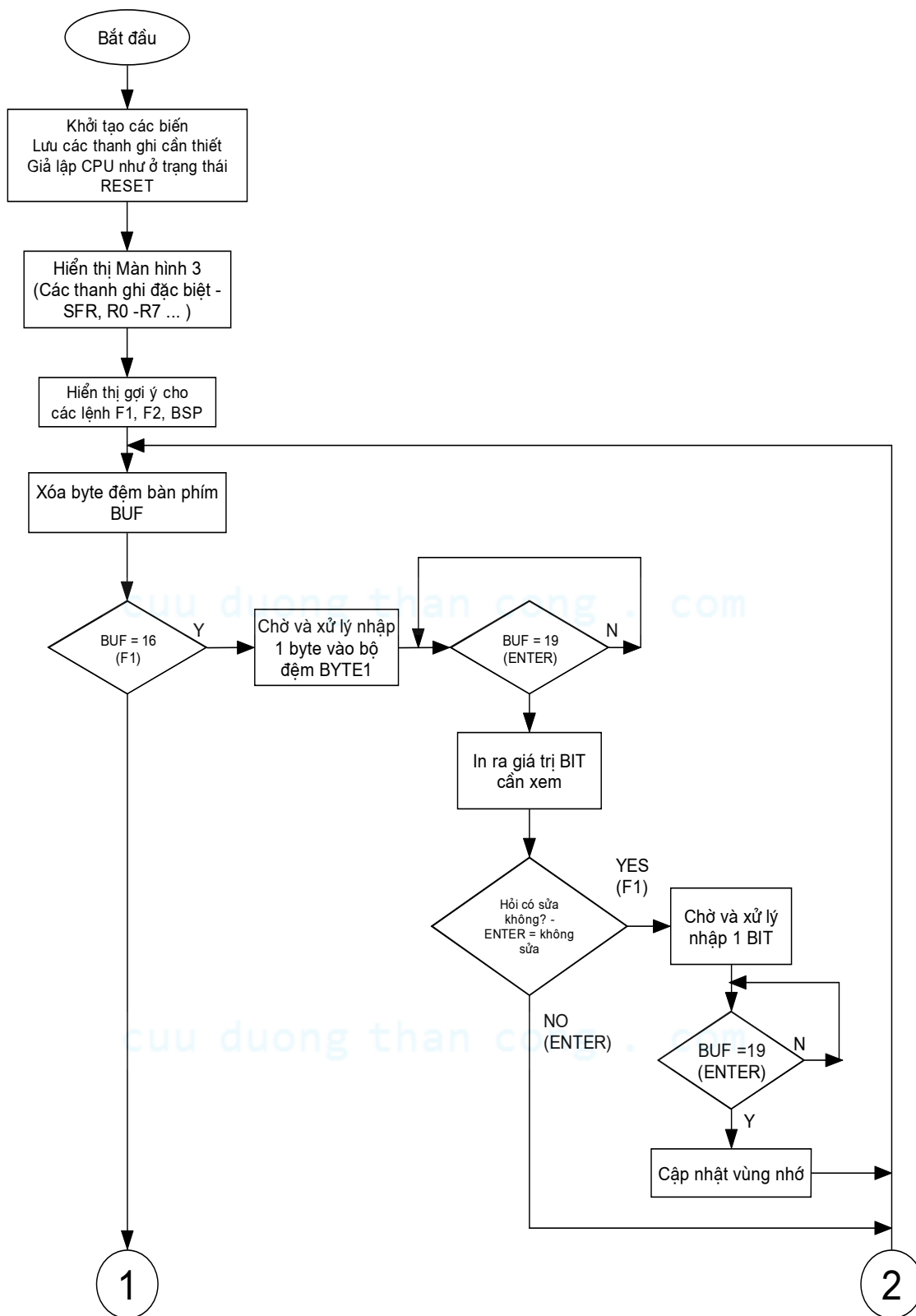
Sau đây là lưu đồ thuật toán của chương trình MONITOR :

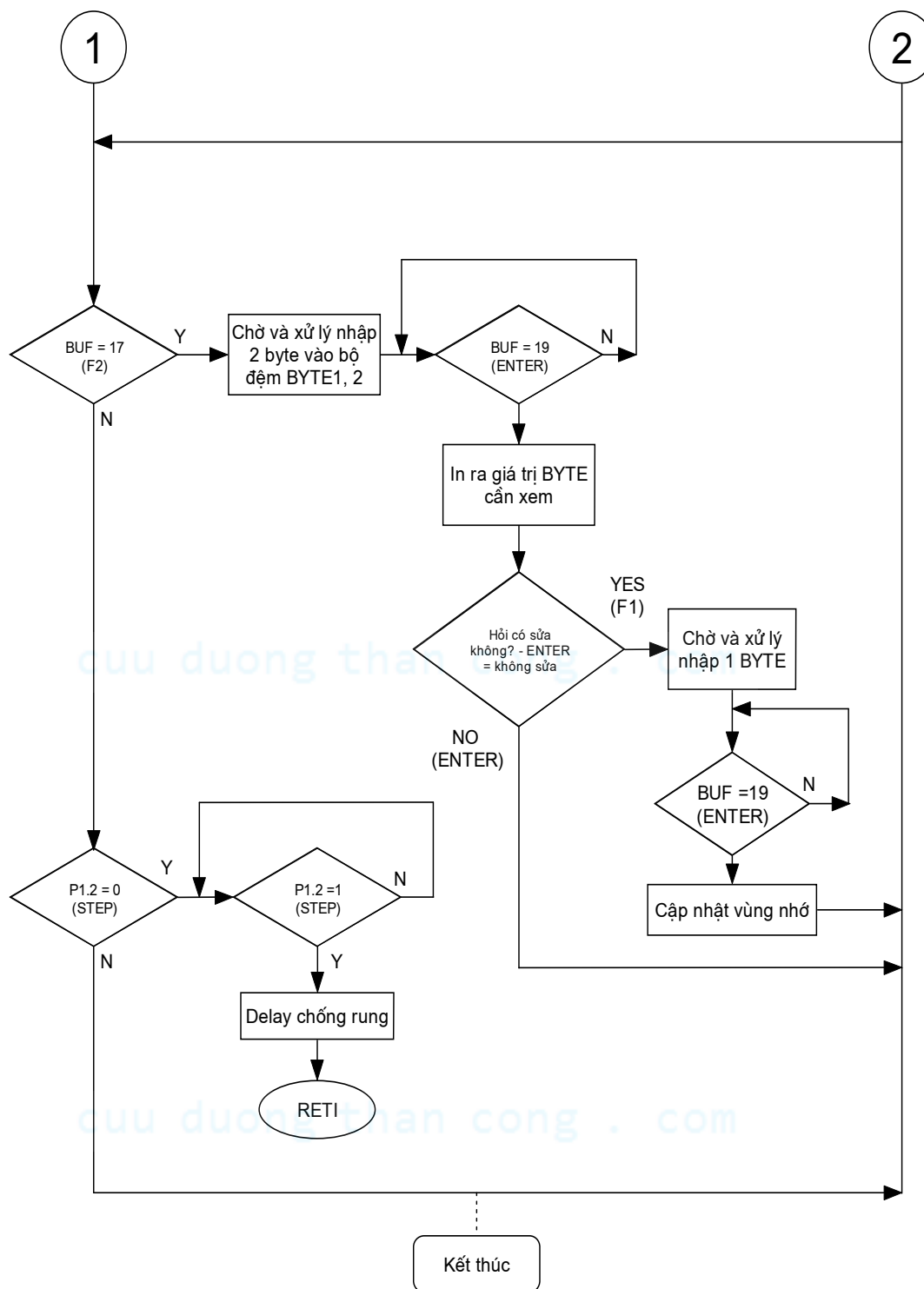
Lưu đồ thuật toán của module Main: khởi tạo các thiết đặt ban đầu cho KIT và các thiết bị ngoại vi, hiển thị màn hình 1.

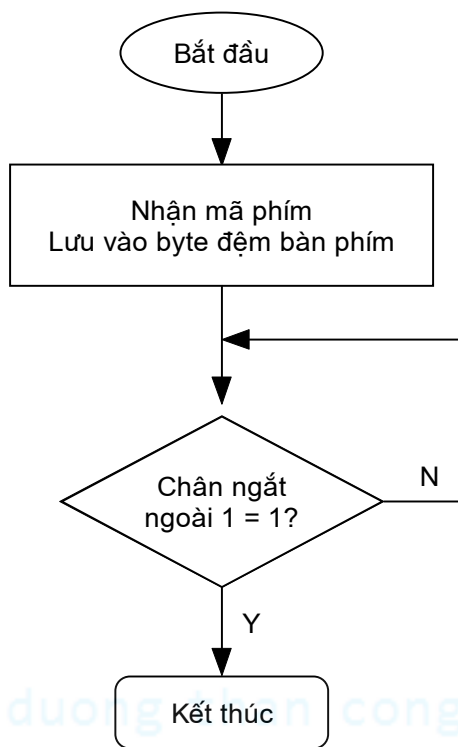


Lưu đồ các chương trình DEMO (Màn hình 4)



Lưu đồ chương trình MONITOR chạy ở chế độ gỡ rối STEP MODE (Màn hình 3)

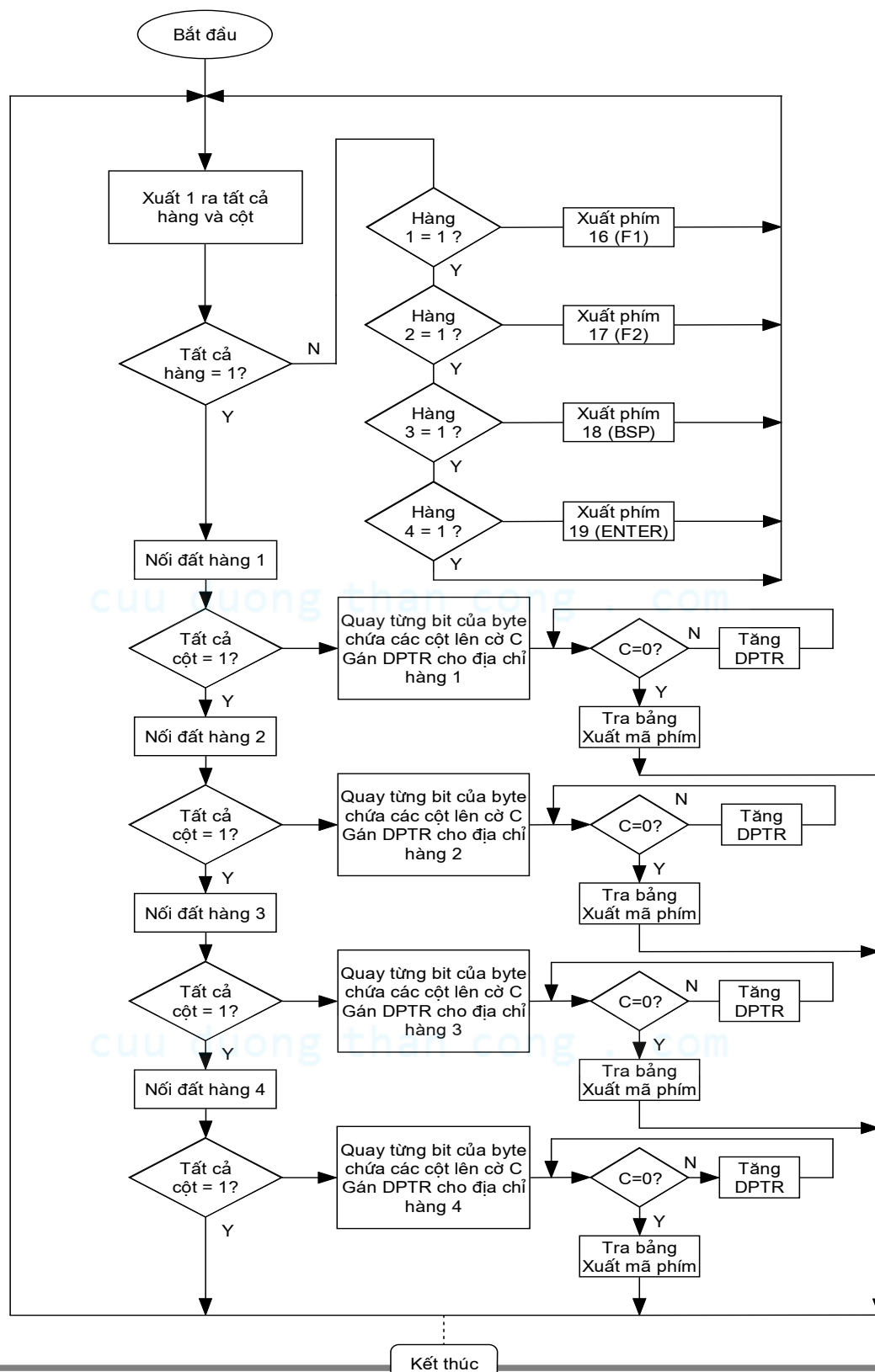


Lưu đồ chương trình ngắt ngoài 0 (INT0) của bàn phím**3.2 Chương trình quản lí thiết bị****3.2.1 Chương trình quét bàn phím**

Như đã trình bày ở chương 2, bàn phím sử dụng trong bộ KIT này gồm có 20 phím, 16 phím đầu tiên được sắp xếp theo kiểu ma trận và 4 phím cuối cùng được xếp theo một cột nối đất chung. Trên bàn phím của bộ KIT, 4 phím trên một cột nối đất chung là các phím chức năng : F1, F2, BACK SPACE và ENTER. Các phím từ 0-9 và A-F dùng để nhập liệu, các phím chức năng có tác dụng khác nhau tùy theo ngữ cảnh của màn hình, gợi ý tác dụng của các phím bấm hiển thị trên màn hình, điều này giúp việc làm quen và thí nghiệm trên KIT rất thuận tiện và dễ dàng cho người học, nhất là khi thí nghiệm ở chế độ debug.

Lưu đồ thuật toán của chương trình quét bàn phím như sau:

Lưu đồ thuật toán của chương trình quét bàn phím:

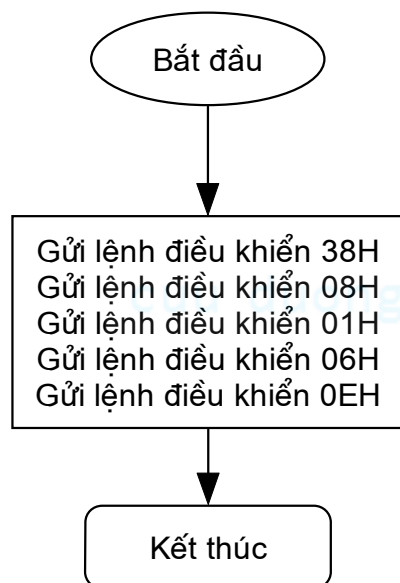


3.2.2 Chương trình điều khiển hiển thị LCD

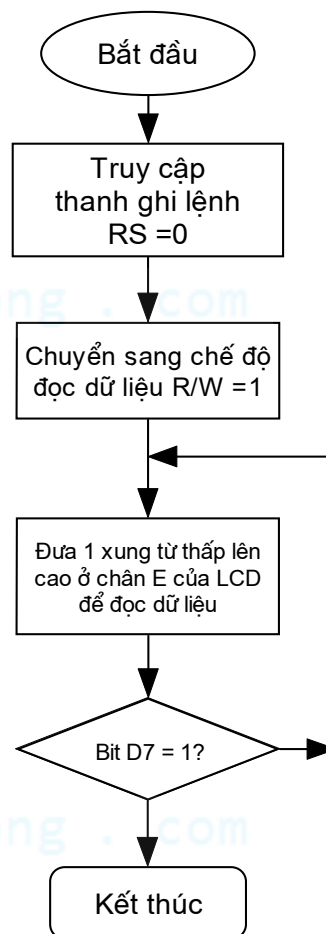
Để cho LCD có thể hoạt động, đầu tiên ta phải khởi tạo LCD, báo cho nó biết số hàng, số ký tự hiển thị trên một hàng. Các dữ liệu điều khiển lần lượt được chuyển vào Data Bus của LCD là 38H, 0EH, và 06H. Sau các lệnh khởi tạo LCD này ta mới có thể hiển thị ký tự lên trên màn hình LCD.

Các lưu đồ làm việc với LCD như sau:

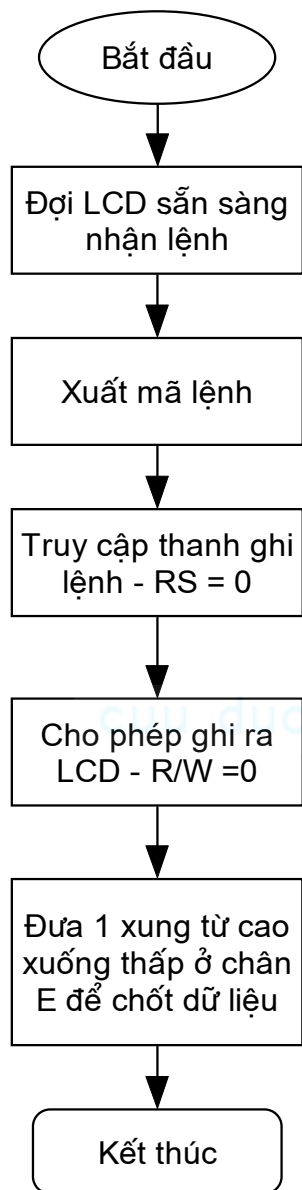
Khởi tạo LCD



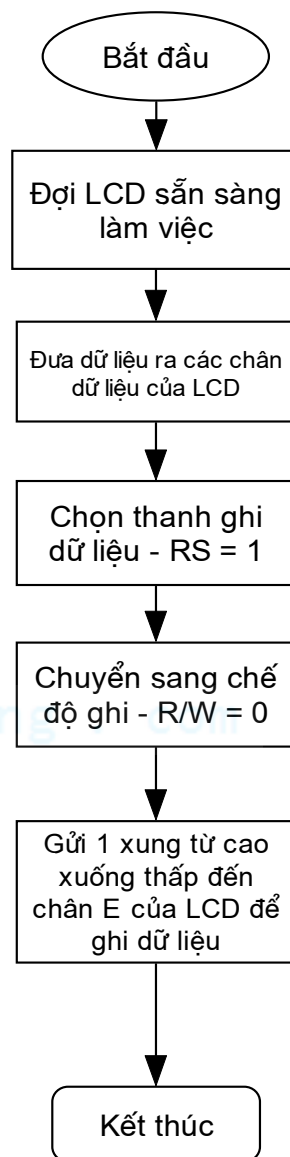
Chờ LCD sẵn sàng làm việc



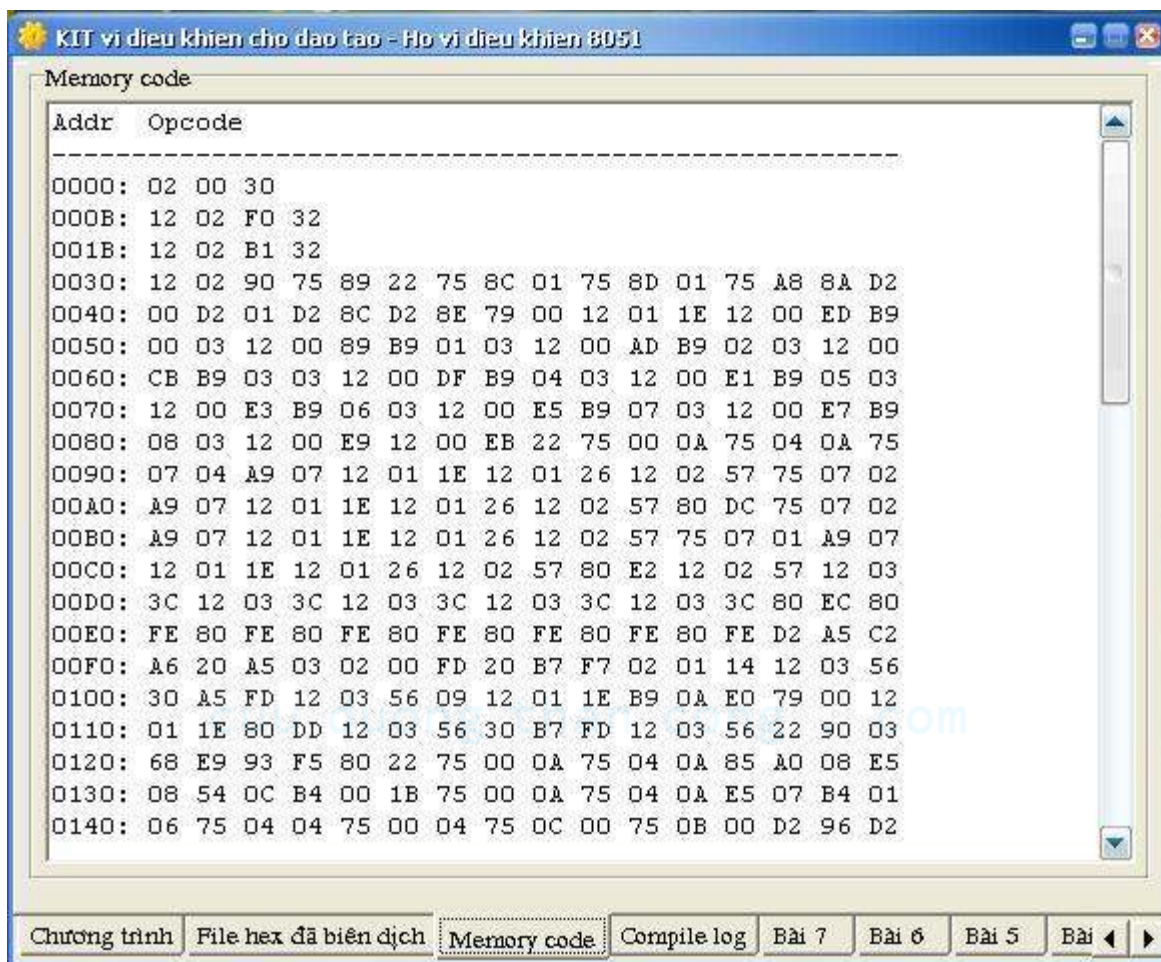
Gửi lệnh điều khiển ra LCD



In một kí tự ra LCD



3.3 Chương trình giao tiếp máy tính



Chương trình giao tiếp giữa bộ KIT với máy tính được viết bằng ngôn ngữ Delphi. Giao diện của chương trình trên máy tính PC như hình vẽ

- Chương trình cho phép nạp chương trình của người sử dụng vào EEPROM, sau đó sẽ chạy trên KIT, có thể chạy độc lập, tức không có sự điều khiển của MONITOR - chế độ STEP hoặc chạy từng lệnh để debug và theo dõi giá trị bộ nhớ, các thanh ghi đặc biệt SFR, R0-R7 trên màn hình LCD của KIT.
- Trên giao diện của chương trình có thể download file dạng Hexa của chương trình xuống bộ KIT và có thể upload nội dung bộ nhớ EEPROM của KIT.
- Các nút chức năng trên giao diện :

+ **OPEN and COMPILE**: Mở một file mã lệnh dạng hợp ngữ (*.asm) và biên dịch (Compile) thành file HEX hoặc file đã compile dưới dạng HEXA (*.hex) để nạp trực tiếp xuống KIT. Kết quả compile được hiển thị trong cửa sổ Compile Log.

+ **OPEN Hex file**: Mở một file mã lệnh đã được biên dịch dạng hexa (*.hex) và dịch ngược (Disassemble) thành file mã lệnh ASM (*.d51). Kết quả dịch ngược hiển thị trong cửa sổ ASM code, mã máy của bộ nhớ hiển thị trong cửa sổ Opcode. Kết quả compile được hiển thị trong cửa sổ Compile Log.

+ **DOWNLOAD** : Cho phép download file Hex được mở hoặc Compile từ file mã lệnh ASM xuống bộ nhớ của KIT.

+ **UPLOAD** : Cho phép đọc nội dung chương trình của người sử dụng đã được nạp vào bộ nhớ của KIT từ lần nạp trước.

+ **Erase EEPROM**: Cho phép xóa EEPROM.

Chương trình MONITOR được viết dưới dạng hợp ngữ (Assembly), ban đầu khi tham khảo có thể gây khó khăn cho người đọc tuy nhiên hợp ngữ làm cho người học có thể hiểu rõ và nắm vững về cấu trúc phần cứng của họ vi điều khiển 8051 hơn như cách tổ chức bộ nhớ, cách thực hiện một ngắt, vị trí của các vector ngắt, các hoạt động định thời, hiểu rõ vị trí bản đồ bộ nhớ, các thanh ghi đặc biệt SFR ... Ngoài ra khi sử dụng hợp ngữ ta không phải phụ thuộc vào các trình biên dịch hỗ trợ bằng ngôn ngữ bậc cao như C, Basic ...

Sau đây ta đi xây dựng các nội dung thí nghiệm cần thiết cho người học để qua đó người học từng bước nắm rõ cấu trúc và cách lập trình cho họ vi điều khiển 8051.

cuuduongthancong.com