

Bài tập dài

Bài tập dài này yêu cầu thiết kế và xây dựng phần mềm hiển thị file .hex, được format theo dạng intel hex file

1. Giới thiệu

Intel hex file là một chuẩn format file hex thường dùng trong việc lưu trữ dữ liệu nhị phân trong các phần mềm làm việc với Vi điều khiển. Cấu trúc của Intel Hex file có thể được tham khảo ở trang sau: <http://www.keil.com/support/docs/1584/>.

Trong các chương trình sử dụng trong vi điều khiển, chương trình dịch (compiler) thường dịch ra file hex (hoặc file .bin), các file này cho phép nói với vi điều khiển là các đoạn mã hex của chương trình sẽ nằm đâu trong bộ nhớ. Để đảm bảo tính chính xác của dữ liệu, Intel Hex luôn đưa ra kiểm tra check sum và nhiệm vụ của lập trình viên là đảm bảo luôn kiểm tra check sum cho mỗi dòng dữ liệu được đọc vào. Theo tiêu chuẩn cũ thì kích thước của file lớn nhất là 64Kbytes, tức là khoảng 65536 ô trong bộ nhớ.

2. Yêu cầu

Chương trình sẽ mở một file vào và tìm hiểu xem file đấy có phải là file hex không? Sau đó chương trình sẽ đọc từng dòng của file và cuối cùng hiển thị file đấy lên trên màn hình. Định dạng của hiển thị như hình vẽ sau, tham khảo file ví dụ với trang Web: <https://hexed.it/>, file (Uart.X.production.hex)

1. Vùng đầu tiên là địa chỉ của ô nhớ, lần lượt đánh số từ 0000 -FFFF
2. Vùng thứ hai là nội dung ô nhớ, dưới dạng hex (các ô nhớ không có dữ liệu sẽ được mặc định là FF)
3. Vùng thứ 3 là giá trị ASCII tương ứng với giá trị hex
4. Vùng nhớ cố định là 64Kbytes, lớn hơn vùng nhớ này sẽ không cần hiển thị.
5. Lưu ý là chương trình chỉ cần hiển thị dạng text đơn giản (không cần phải format màu như hình ví dụ).
6. Các file .hex mẫu được đính kèm với yêu cầu này.

-Untitled- x	Uart.X.production.hex x		
00000000	00 02 04 00 00 00 00 00	52 03 00 00 52 03 00 00 R . . R . .
00000010	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000020	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000030	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000040	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000050	52 03 00 00 C8 02 00 00	52 03 00 00 52 03 00 00	R . . ℒ . . R . . R . .
00000060	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000070	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000080	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000090	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
000000A0	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
000000B0	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
000000C0	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
000000D0	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
000000E0	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
000000F0	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000100	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000110	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000120	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000130	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000140	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000150	52 03 00 00 FF FF FF FF	FF FF FF FF FF FF FF FF	R . .
00000160	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
00000170	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
00000180	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
00000190	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
000001A0	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
000001B0	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
000001C0	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
000001D0	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
000001E0	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
000001F0	FF FF FF FF FF FF FF FF	FF FF FF FF FF FF FF FF	
00000200	FF FF FF FF FF FF FF FF	52 03 00 00 52 03 00 00	R . . R . .
00000210	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000220	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000230	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000240	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000250	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .
00000260	52 03 00 00 52 03 00 00	52 03 00 00 52 03 00 00	R . . R . . R . . R . .

3. Yêu cầu kỹ thuật

Chương trình sẽ bao gồm các chức năng sau:

- Chương trình sẽ chạy từ dòng lệnh (command – line), với tên file chạy là HexDisplay, sau đấy tên của file đầu vào
- Chương trình sẽ kiểm tra dạng xem có phải là file hex không, nếu không phải thì sẽ in ra thông báo lỗi

Sau khi đọc xong dữ liệu vào bộ nhớ, chương trình sẽ in nội dung màn hình, lưu ý mỗi trang màn hình sẽ có độ dài là 25 dòng, nếu chương trình dài hơn 25 dòng thì phần mềm hỏi người sử dụng có tiếp tục hiển thị nữa không, nếu có thì chương trình sẽ tự động xóa đi 25 dòng này và hiển thị 25 dòng tiếp theo, điều này tiếp tục cho khi đến hết số dòng cần hiển thị

Xử lý lỗi

1. Usage: HexDisplay <input file>

Nếu một file mà không mở được thì sẽ có thông báo
Error: FILENAME could not be opened.
với tên file là file không mở được

Nếu trong chương trình có bất cứ dòng nào mà không đúng format của file Intel hex, thì chương trình sẽ thông báo lỗi tại dòng đó và kết thúc.

5 Thiết kế

Phương pháp thiết kế truyền thống (top – down). Bạn sẽ phải vẽ sơ đồ thiết kế hình cây và đưa ra giải thích về sự chuyển đổi thông tin giữa các phần của chương trình.

6 Code

Lưu ý, để dễ dàng cho việc chấm, mỗi hàm cần có lời giải thích ngắn gọn, đầu vào là gì, đầu ra là gì . Bạn nên đưa ra giải thích trong hàm khi có thể. Lưu ý việc sử dụng tên hàm, tên biến cho dễ hiểu và thống nhất. Các hàm phải cách nhau bằng hàng trống, tránh sử dụng các giá trị hằng.

7. Gợi ý

Sử dụng các hàm sau khi làm việc với file fgetc(), fgets(), ungetc(), fputc(), fprintf(), and rewind().

Sử dụng vùng nhớ đệm, ví dụ buffer[65536];

Sử dụng hàm đọc file để đọc dữ liệu vào bộ nhớ đệm

Sau khi đọc xong mới tiến hành gửi dữ liệu này ra màn hình.

Yêu cầu bài tập dài số

1. Thời hạn nộp bài tập dài: Nộp bài vào trang assignment của Team.
2. Phần mềm nộp được nộp bao gồm
 - a. **Báo cáo về bài tập dài**
 - b. Chương trình phần mềm chạy bài tập gồm file mã nguồn và file chạy chương trình (.exe) viết trên Win32.
 - c. Mỗi thư mục ghi rõ: Họ và tên sinh viên, lớp và số hiệu SV
3. Bài tập dài sẽ được đánh giá trên hai điểm chính:
 - a. Chương trình chạy có đáp ứng được các yêu cầu bài tập đặt ra không?
 - b. Mã nguồn của chương trình có được viết tốt không?
4. Nội dung của báo cáo cho bài tập dài số bao gồm những phần sau:
 - a. Những yêu cầu nào mà phần mềm của anh (chị) sẽ giải quyết, bao gồm các đầu vào của chương trình là thế nào ? Chương trình sẽ không hoạt động khi cho vào những dữ liệu kiểu thế nào ? (Lưu ý không tìm cách thêm chức năng cho phần mềm).
 - b. Cấu trúc cơ bản của phần mềm viết ra theo sơ đồ hình cây (top down), đồng thời giải thích rõ các cấu trúc dữ liệu chính trong phần mềm .
 - c. Những điểm theo anh (chị) là quan trọng quyết định đến hiệu quả của phần mềm này
 - d. Bản báo cáo sẽ dài tối đa là 10 trang (ngoại trừ hình vẽ) và không bao gồm mã nguồn của chương trình.
5. Lưu ý
 - a. Đây là bài tập dài dành cho từng cá nhân và mọi người phải tự làm toàn bộ các báo cáo và phần mềm của riêng mình. Mọi sao chép (dù chỉ một phần) trong báo cáo hay trong phần mềm đều bị coi là phạm quy và bài viết sẽ được cho điểm 0.