

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG



BÁO CÁO

BÀI TẬP LỚN VI XỬ LÝ

ĐỀ TÀI:

**THIẾT KẾ MẠCH ĐỒNG HỒ LỊCH VẠN NIÊN
SỬ DỤNG VI ĐIỀU KHIỂN AVR**

Giảng viên hướng dẫn: TS. Nguyễn Hoàng Dũng

Sinh viên thực hiện: Hoàng Tuấn Linh – 20132263

Nguyễn Văn Long – 20132393

Lớp: KSTN – ĐTVT – K58

Hà Nội, 4-2016

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG



BÁO CÁO

BÀI TẬP LỚN VI XỬ LÝ

ĐỀ TÀI:

**THIẾT KẾ MẠCH ĐỒNG HỒ LỊCH VẠN NIÊN
SỬ DỤNG VI ĐIỀU KHIỂN AVR**

Giảng viên hướng dẫn: TS. Nguyễn Hoàng Dũng
Sinh viên thực hiện: Hoàng Tuấn Linh – 20132263
Nguyễn Văn Long – 20132393
Lớp: KSTN – ĐTVT – K58

Nhận xét của giáo viên

This image shows a full page of white paper with horizontal dashed lines. The lines are evenly spaced and run across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

Mục lục

Nhận xét của giáo viên.....	3
Danh mục hình vẽ	5
Danh mục bảng biểu	6
Lời mở đầu.....	7
Chương I. Mô tả đề tài.....	8
1. Đặt vấn đề	8
2. Phân tích yêu cầu thiết kế.....	9
2.1 Yêu cầu chức năng	9
2.2 Yêu cầu phi chức năng.....	9
Chương II. Lập kế hoạch.....	10
1. Bảng phân tích nhân lực.....	10
2. Bảng phân công công việc	11
3. Bảng kế hoạch thực hiện.....	12
Chương III. Thiết kế hệ thống.....	13
1. Thiết kế sơ đồ khối.....	13
2. Thiết kế sơ đồ khối chi tiết.....	14
2.1 Khối nguồn.....	14
2.2 Khối điều chỉnh ngày giờ và báo thức	14
2.3 Khối hiển thị.....	15
2.4 Khối thời gian thực	16
2.5 Khối điều khiển.....	18
3 Sơ đồ nguyên lý	20
Chương IV. Hoàn thành sản phẩm.....	21
1. Layout mạch in.....	21
2. Làm mạch, hàn linh kiện.....	22
3. Đo đạc, kiểm tra, đánh giá chất lượng sản phẩm	23
Chương VI. Kết luận.....	25
1. Đánh giá những điều đã làm được và chưa làm được	25
1.1 Những điều đã làm được	25
1.2 Những điểm còn hạn chế.....	25
2. Hướng phát triển	26
3. Kết luận	26
Tài liệu tham khảo.....	27
Phụ lục 1: Source code.....	28
Phụ lục 2: Hướng dẫn sử dụng.....	44

Danh mục hình vẽ

Hình 1. Sơ đồ khối tổng quan mạch đồng hồ lịch vạn niên	13
Hình 2. Adapter 5V – 2A	14
Hình 3 Khối điều chỉnh ngày giờ và báo thức	15
Hình 4. Nút bấm	15
Hình 5. Điện trở	15
Hình 6. Khối hiển thị.....	16
Hình 7. LCD 16*2	16
Hình 8. Khối thời gian thực	17
Hình 9. Thạch anh 32,768MHz	17
Hình 10. IC DS1307	17
Hình 11. Battery 3V.....	17
Hình 12. Khối điều khiển	18
Hình 13. VĐK ATmega16	18
Hình 14. Sơ đồ chân VĐK ATmega16	19
Hình 15. Sơ đồ nguyên lý	20
Hình 16. Mạch in PCB 2D	21
Hình 17. Mạch in PCB 3D	21
Hình 18. Kiểm tra mạch trên board	22
Hình 19. Mạch in chưa hàn linh kiện.....	22
Hình 20. Hình ảnh mạch thực tế đang hoạt động.....	23
Hình 21. Mặt sau mạch hoàn chỉnh	23

Danh mục bảng biểu

<i>Bảng 1. Bảng phân tích nhân lực.....</i>	<i>10</i>
<i>Bảng 2. Bảng phân công công việc.....</i>	<i>11</i>
<i>Bảng 3. Bảng kế hoạch thực hiện.....</i>	<i>12</i>

Lời mở đầu

Với sự phát triển không ngừng của khoa học kỹ thuật, đặc biệt là ngành điện tử đã được ứng dụng rất nhiều trong công nghiệp. Trong lĩnh vực điều khiển, từ khi công nghệ chế tạo loại vi mạch lập trình phát triển đã đem đến các kỹ thuật điều khiển hiện đại có nhiều ưu điểm hơn so với việc sử dụng các mạch điều khiển lắp ráp bằng các linh kiện rời như kích thước nhỏ, giá thành rẻ, độ làm việc tin cậy, công suất tiêu thụ nhỏ.

Ngày nay, trong lĩnh vực điều khiển đã được ứng dụng rộng rãi trong các thiết bị, sản phẩm phục vụ cho nhu cầu sinh hoạt hàng ngày của con người như máy giặt, đồng hồ báo giờ... đã giúp cho đời sống của chúng ta ngày càng hiện đại và tiện nghi hơn.

Chúng em đã chọn đề tài “Thiết kế mạch đồng hồ lịch vạn niên sử dụng vi điều khiển AVR” để hoàn thành bài tập lớn môn Vi xử lý, cũng như để đáp ứng mong muốn tìm hiểu và làm một sản phẩm điện tử cụ thể từ một dòng Vi điều khiển của bản thân.

Đề tài “Thiết kế mạch đồng hồ lịch vạn niên” là một đề tài quen thuộc, có thể thực hiện trên rất nhiều dòng VDK. Nhưng đối với chúng em, khi mới tìm hiểu một dòng VDK mới, và đặc biệt phải thiết kế mạch đồng hồ trên dòng VDK AVR theo yêu cầu, đó cũng là một thách thức và trải nghiệm rất thú vị. Do kiến thức còn hạn chế nên đề tài chắc chắn còn nhiều thiếu sót. Chúng em rất mong nhận được những ý kiến đóng góp để chúng em hoàn thiện đề tài từ phía thầy cô cũng như các bạn sinh viên.

Chúng em xin chân thành cảm ơn thầy Nguyễn Hoàng Dũng và các nhóm trợ giảng đã giúp đỡ chúng em rất nhiều trong tất cả quá trình, từ lúc mới bắt đầu đến lúc hoàn thiện sản phẩm!

Chương I. Mô tả đề tài

1. Đặt vấn đề

Bài toán đặt ra là thiết kế mạch đồng hồ lịch vạn niên sử dụng VDK Atmega16 là VDK chính. Đây là bài toán khá quen thuộc và có rất nhiều cách giải quyết bài toán này. Đề tài sử dụng IC thời gian gian thực DS1307, có tác dụng thực hiện việc đếm lên ngày tháng, VDK AVR chỉ thực hiện đặt ngày giờ để đưa vào DS1307, sau đó thực hiện đọc ngày giờ từ DS1307 ra. VDK còn có chức năng nhận các tín hiệu đặt ngày giờ và đặt báo thức từ các nút bấm, sau đó xử lý để đưa ra ngày giờ mong muốn rồi gửi dữ liệu vào DS1307. Đề hiện thị có 2 lựa chọn là sử dụng LED 7 đoạn và sử dụng LCD. Đề tài đã chọn phương án sử dụng LCD để có thể hiển thị nhiều thông tin so với LED 7 đoạn, hơn nữa việc triển khai lại có phần dễ dàng hơn (nếu sử dụng LED 7 đoạn thì phải tính toán thời gian quét LED).

Từ việc phân tích và khảo sát trên đây, nhóm đã đưa ra các vấn đề chính cần giải quyết của bài toán đặt ra là:

- Tìm hiểu kiến thức cơ bản về VDK Atmega16, đặc biệt về phần giao tiếp với DS1307 và LCD. Tìm hiểu về phần mềm CodeVisionAVR C Compiler để thực hiện code cho VDK Atmega16.
- Xây dựng sơ đồ khối, từ đó xây dựng mạch nguyên lý và mô phỏng trên phần mềm Proteus.
- Thiết kế mạch in trên phần mềm Altium Designer.
- Thực hiện lắp ráp linh kiện trên mạch in. Để nạp code cho VDK nhóm sử dụng phần mềm Progisp và mạch nạp USB ISP.
- Viết báo cáo tổng hợp về quy trình thực hiện đề tài.

2. Phân tích yêu cầu thiết kế

2.1 Yêu cầu chức năng

Mạch có các yêu cầu chức năng sau:

- Hiển thị giờ - phút – giây trên dòng thứ nhất của LCD 16*2
- Hiển thị thứ - ngày - tháng – năm trên dòng thứ 2 của LCD 16*2
- Có thể đặt lại ngày – giờ cũng như thứ - ngày – tháng – năm nhờ vào 3 phím bấm MODE, UP, DOWN
- Có chức năng báo thức, thực hiện bằng đèn và còi báo hiệu. Có thể tắt báo thức bằng nút bấm OK, khi đó nếu muốn báo thức lại phải cài đặt lại ngày giờ báo thức rồi xác nhận bằng phím OK. Nếu không tắt báo thức thì đèn và còi sẽ tự động hủy báo thức sau 1 phút, trong trường hợp này báo thức vẫn sẽ hoạt động vào ngày hôm sau.

2.2 Yêu cầu phi chức năng

Mạch có các yêu cầu phi chức năng sau:

- Sử dụng nguồn 1 chiều 5V
- Kích thước mạch thực tế: 5cm * 7cm
- Sử dụng VĐK PIC là VĐK chính
- Hiển thị trên LCD 16*2
- Sản phẩm được thiết kế phù hợp với việc đặt cố định, lựa chọn sử dụng linh kiện sao cho mức giá thấp nhất có thể
- Tổng chi phí dự kiến: 300.000VNĐ

Chương II. Lập kế hoạch

Quá trình lập kế hoạch giúp chúng em có cái nhìn tổng quan về mục tiêu, nhiệm vụ cần thực hiện cũng như phân bổ thời gian hợp lý cho toàn bộ quá trình thiết kế sản phẩm. Đây là bước giúp chúng em hiểu hơn về từng thành viên trong nhóm, từng công việc cụ thể cần phải thực hiện.

1. Bảng phân tích nhân lực

Bảng 1. Bảng phân tích nhân lực

Thành viên	Ưu điểm	Hạn chế
Hoàng Tuấn Linh	<ul style="list-style-type: none"> - Kỹ năng thuyết trình báo cáo tốt, tự tin. - Có kiến thức cơ bản về điện tử, các loại vi điều khiển. - Thành thạo lập trình cho Vi điều khiển AVR - Kỹ năng hàn mạch, test mạch tốt, cẩn thận. 	Chưa thành thạo các công cụ mô phỏng, thiết kế điện tử.
Nguyễn Văn Long	<ul style="list-style-type: none"> - Có kiến thức cơ bản về điện tử. - Kỹ năng làm Slide, báo cáo Word Excel tốt. - Kỹ năng thiết kế mạch in nhanh chóng, đẹp. - Kỹ năng lập kế hoạch, quản lý đảm bảo tiến độ công việc. 	Hạn chế về Code, thuật toán xử lý cho Vi điều khiển.

2. Bảng phân công công việc

Bảng 2. Bảng phân công công việc

Nội dung	Công việc	Thành viên
Tìm hiểu đề tài	Xác định đề tài, yêu cầu thiết kế của Mạch đồng hồ Lịch vạn niên.	Tuấn Linh
	Tìm hiểu kiến thức về Vi điều khiển AVR, chip Atmega16.	Tuấn Linh + Văn Long
	Lập kế hoạch chi tiết, phân công công việc cho quá trình thiết kế sản phẩm.	Văn Long
Thiết kế hệ thống	Thiết kế sơ đồ khối	Tuấn Linh
	Thiết kế chi tiết từng khối	Tuấn Linh + Văn Long
	Vẽ mạch nguyên lý và mô phỏng trên Proteus.	Tuấn Linh
	Lập trình cho Vi điều khiển Atmega16	Tuấn Linh + Văn Long
Hoàn thành sản phẩm	Thiết kế mạch in, đặt mạch in.	Văn Long
	Mua linh kiện, test nguội mạch in	Văn Long
	Hàn linh kiện.	Tuấn Linh
	Đo đạc, kiểm tra, đánh giá sản phẩm	Tuấn Linh + Văn Long
Báo cáo bài tập lớn	Hoàn thành nội dung báo cáo Word, video giới thiệu sản phẩm.	Tuấn Linh + Văn Long
	Thuyết trình, báo cáo trước giảng viên.	Tuấn Linh + Văn Long

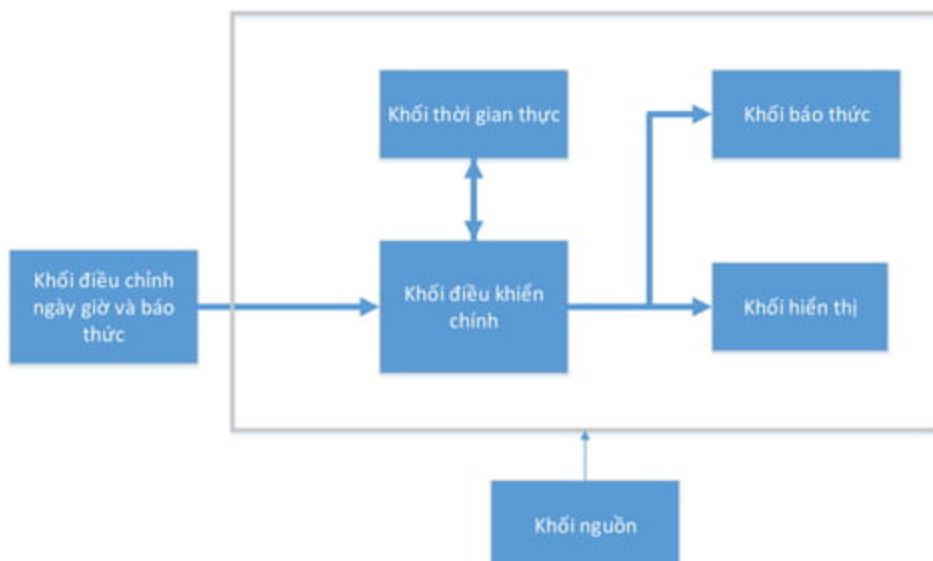
3. Bảng kế hoạch thực hiện

Bảng 3. Bảng kế hoạch thực hiện

Công việc	Thời gian bắt đầu	Thời gian kết thúc	Yêu cầu đạt được	Trạng thái
Xác định đề tài Mạch đồng hồ Lịch vạn niên.	10/02	17/02	Xác định được yêu cầu thiết kế về chức năng và phi chức năng của mạch	Hoàn thành
Tìm hiểu về Vi điều khiển AVR, code cho AVR	18/02	01/03	Hiểu và trình bày được những kiến thức về Vi điều khiển AVR	Hoàn thành
Thiết kế mạch nguyên lý.	02/03	16/03	Hoàn thành sơ đồ khối chi tiết, mạch nguyên lý	Hoàn thành
Viết Code, mô phỏng mạch trên Proteus	17/03	02/05	Mô phỏng và chạy đúng với yêu cầu thiết kế trên phần mềm mô phỏng Proteus	Hoàn thành
Thiết kế mạch in	03/05	10/05	Hoàn thành thiết kế mạch in trên phần mềm Altium	Hoàn thành
Hoàn thành, kiểm tra, đánh giá sản phẩm	11/05	18/05	Sản phẩm hoàn thành, chạy đúng yêu cầu thiết kế đề ra	Hoàn thành
Kiểm tra, hoàn thiện báo cáo	19/05	03/06	Bản báo cáo Bài tập lớn, video giới thiệu sản phẩm.	Hoàn thành

Chương III. Thiết kế hệ thống

1. Thiết kế sơ đồ khối



Hình 1. Sơ đồ khối tổng quan mạch đồng hồ lịch vạn niên

Hình 1 biểu diễn sơ đồ khối tổng quan của toàn bộ đề tài mạch đồng hồ lịch vạn niên. Có 6 khối chính như trong hình vẽ:

- Khối nguồn sẽ cung cấp nguồn vào cho các khối: Khối thời gian thực, Khối báo thức, Khối hiển thị và Khối điều khiển chính.
- Khối điều chỉnh ngày giờ và báo thức là 4 nút bấm MODE, UP, DOWN và OK.
- Khối điều khiển chính sẽ nhận tín hiệu từ Khối điều chỉnh ngày giờ và báo thức, sau đó thực hiện xử lý tín hiệu rồi giao tiếp với Khối thời gian thực để cài đặt và gọi thời gian. Song song với quá trình đó, Khối điều chỉnh cũng gửi dữ liệu đến các Khối báo thức và Khối hiển thị. Khối điều khiển chính là VXL ATmega16.

- Khối hiển thị nhận tín hiệu từ Khối điều khiển để hiển thị các thông tin về ngày- giờ và thứ - ngày – tháng – năm như yêu cầu chức năng. Khối hiển thị chính là LCD 16*2.
- Khối báo thức gồm 1 đèn LED báo hiệu và một còi chip. Khi nhận được tín hiệu từ VĐK, đèn và còi sẽ hoạt động để thực hiện chức năng báo thức.

2. Thiết kế sơ đồ khối chi tiết

2.1 Khối nguồn

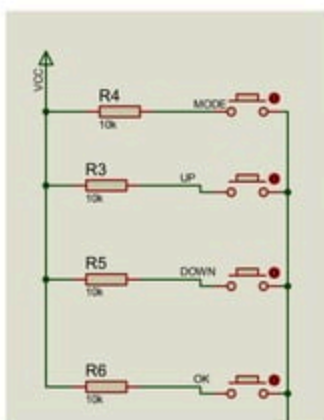


Hình 2. Adapter 5V – 2A

Mạch sử dụng nguồn lấy từ adapter 5V-2A như trong hình 2 để cung cấp đủ dòng cho toàn bộ các khối.

2.2 Khối điều chỉnh ngày giờ và báo thức

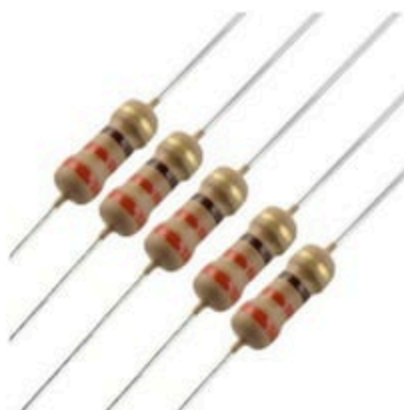
Mạch sử dụng các nút bấm như trong hình 4 để điều chỉnh ngày giờ và đặt báo thức:



Hình 3 Khôi điều chỉnh ngày giờ và báo thức



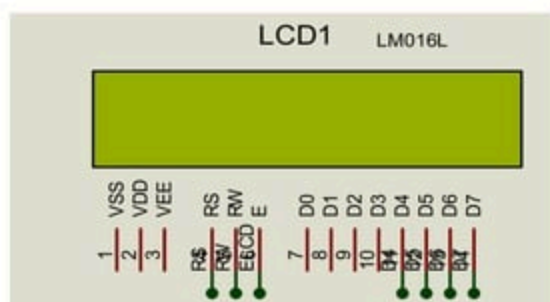
Hình 4. Nút bấm



Hình 5. Điện trở

2.3 Khôi hiển thị

Khôi hiển thị sử dụng LCD 16*2. LCD có 16 chân. Ta cấp nguồn cho LCD thông qua các chân 1 (VSS) và 2 (VDD). Điều chỉnh độ sáng của LCD thông qua các chân 3 (Contrast Voltage), 15 (Backlight Anode) và 16 (Backlight Cathode). Các chân 7 đến 14 là các chân dữ liệu được nối với VĐK, với đề tài ta chỉ sử dụng các chân dữ liệu từ 11 đến 14.



Hình 6. Khởi hiển thị

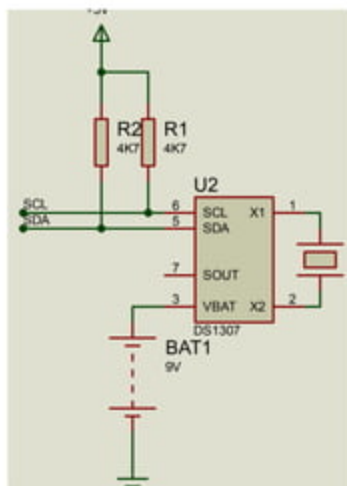


Hình 7. LCD 16*2

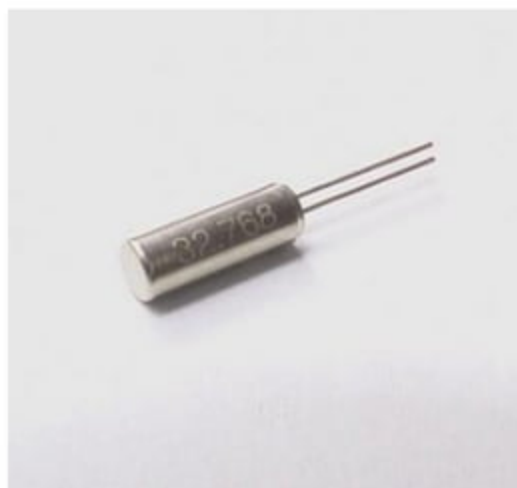
2.4 Khởi thời gian thực

Khởi thời gian thực sử dụng IC DS1307. Ta cấp nguồn cho DS1307 qua trở kéo lên $4.7k\Omega$ vào chân số 4 và 8 của IC. Ngoài ra nguồn pin 3V được nối vào chân 3 là nguồn pin dự trữ khi rút nguồn cấp chính, đảm bảo IC vẫn hoạt động (lúc khởi động lại mạch không bị sai giờ khi mất nguồn nuôi bất chợt). Chân 1 và 2 của IC nối với

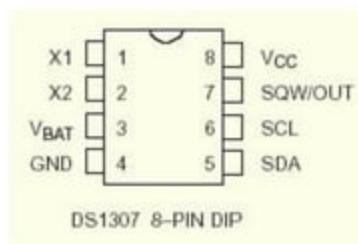
thạch anh 32,768 để tạo dao động. Hai chân 5 và 6 của IC là chân SCL (Serial Clock) và SDA (Serial Data), nối với 2 cổng của VĐK, các chân này sẽ gửi clock và gửi/nhận dữ liệu cho VĐK.



Hình 8. Khối thời gian thực



Hình 9. Thạch anh 32,768MHz

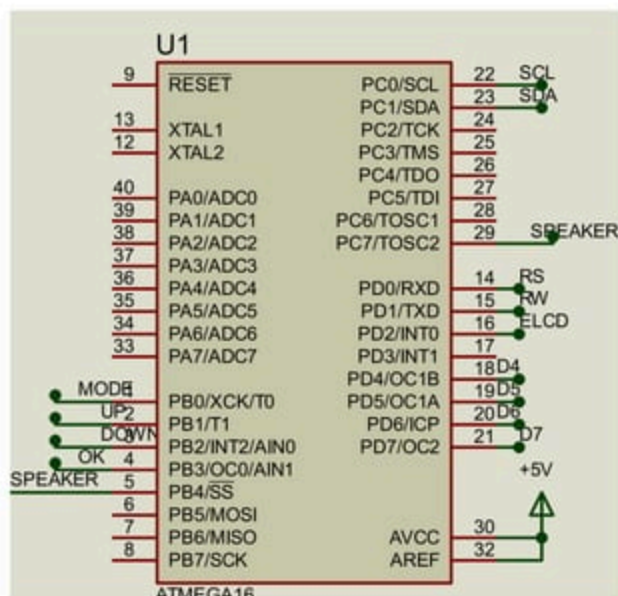


Hình 10. IC DS1307



Hình 11. Battery 3V

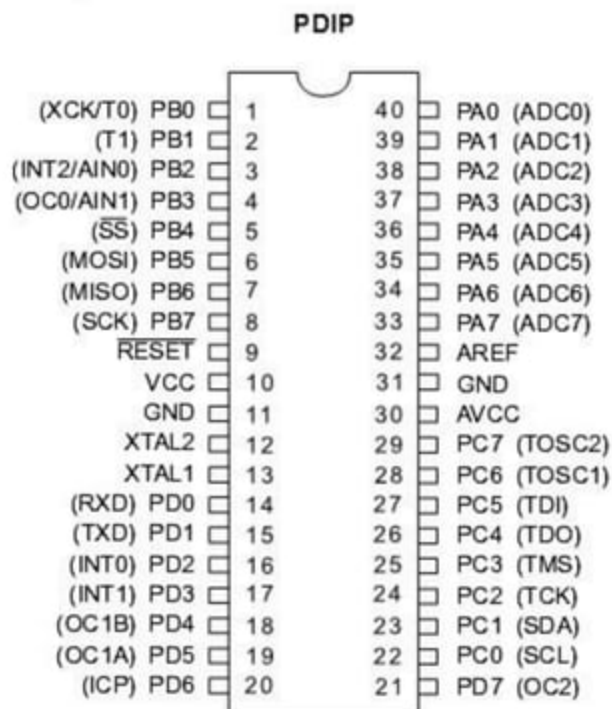
2.5 Khởi điều khiển



Hình 12. Khởi điều khiển



Hình 13. VDK ATmega16



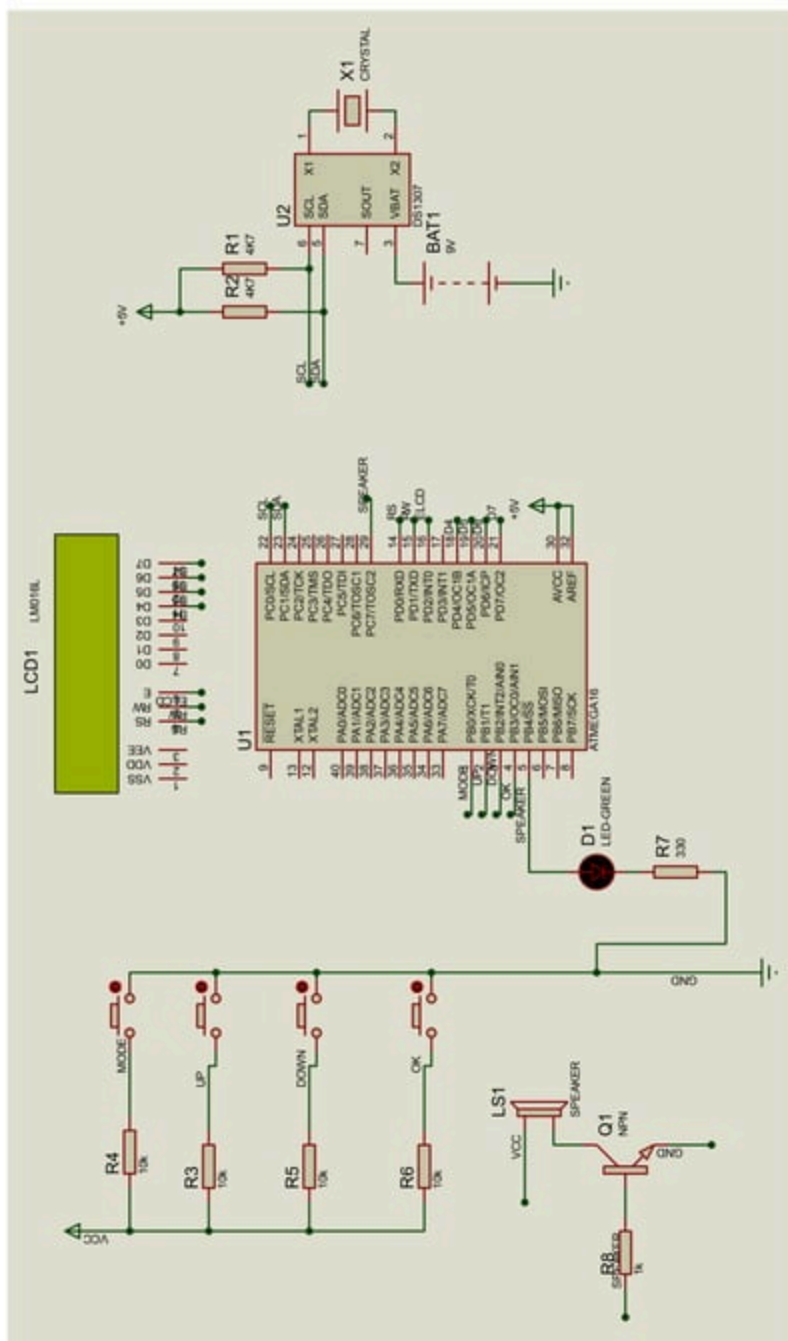
Hình 14. Sơ đồ chân VĐK ATmega16

Khối điều khiển chính là VĐK Atmega16. VĐK này có 40 chân, sơ đồ các chân như trong hình 14.

ATmega16 là một loại Vi điều khiển có nhiều tính năng đặc biệt thích hợp cho việc giải quyết những bài toán điều khiển trên nền vi xử lý.

ATmega16 là vi điều khiển 8bit dựa trên kiến trúc RISC. Với khả năng thực hiện mỗi lệnh trong vòng một chu kỳ xung clock, Atmega16 có thể đạt được tốc độ 1MIPS trên mỗi MHz (1 triệu lệnh/s/MHz), các lệnh được xử lý nhanh hơn, tiêu thụ năng lượng thấp.

3 Sơ đồ nguyên lý

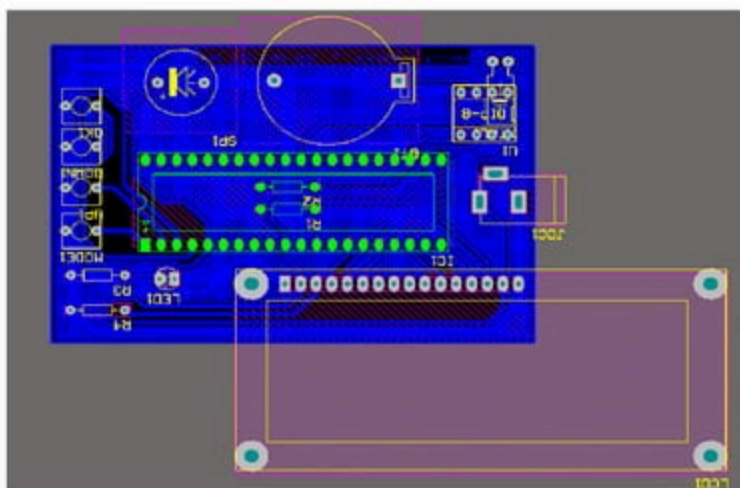


Hình 15. Sơ đồ nguyên lý

Chương IV. Hoàn thành sản phẩm

Quá trình Hoàn thành sản phẩm là bước cuối cùng trong toàn bộ quy trình thiết kế một sản phẩm điện tử.

1. Layout mạch in



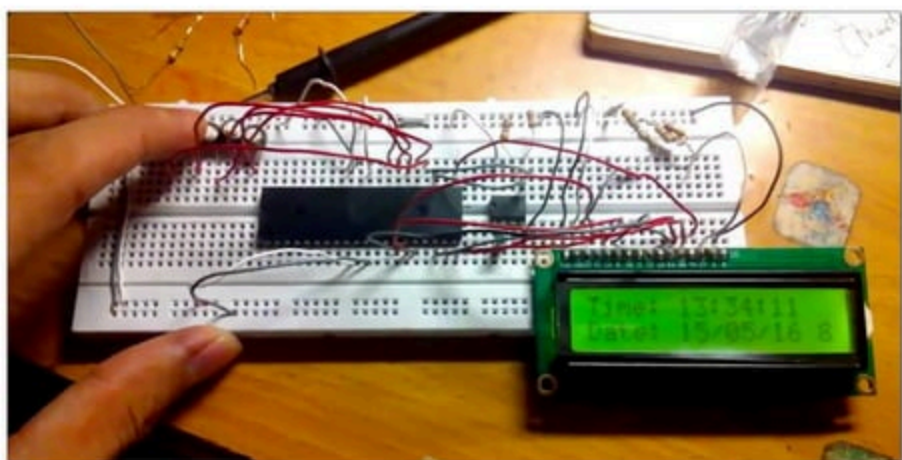
Hình 16. Mạch in PCB 2D



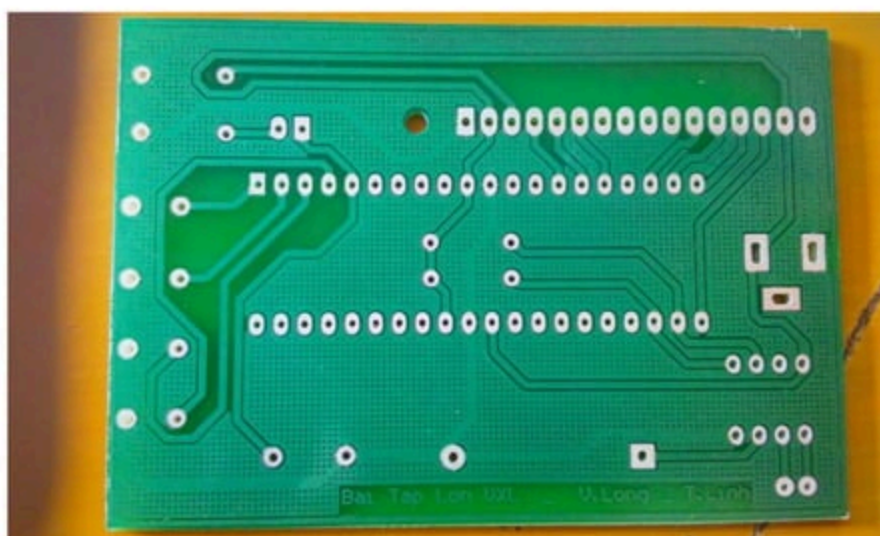
Hình 17. Mạch in PCB 3D

2. Làm mạch, hàn linh kiện

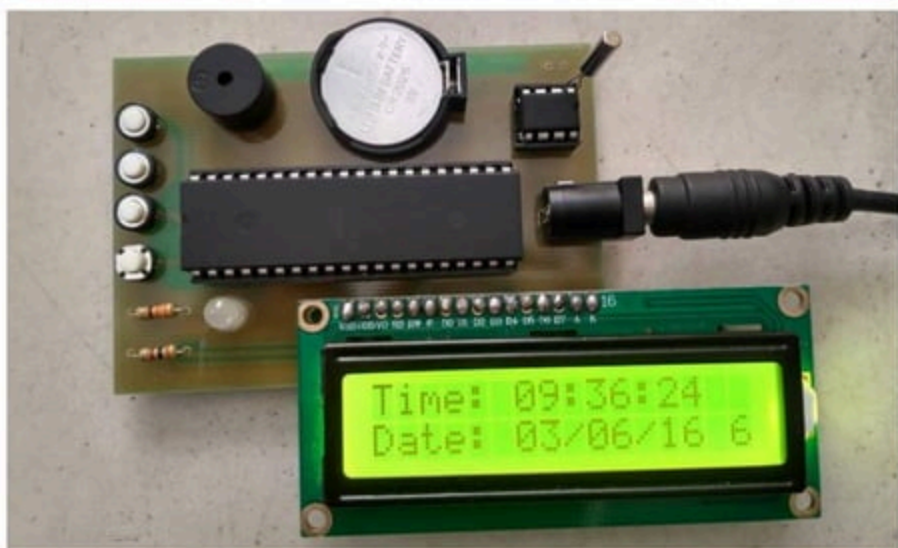
Sau khi có mạch in và các linh kiện đầy đủ, ta sẽ tiến hành lắp ráp thành mạch hoàn chỉnh. Sử dụng mạch nạp và phần mềm nạp để kết nối giữa máy tính và mạch đã lắp ráp. Tùy từng mạch nạp mà có phần mềm nạp riêng. Sau khi nạp xong chúng ta có thể test mạch thử và cung cấp nguồn cho mạch hoạt động.



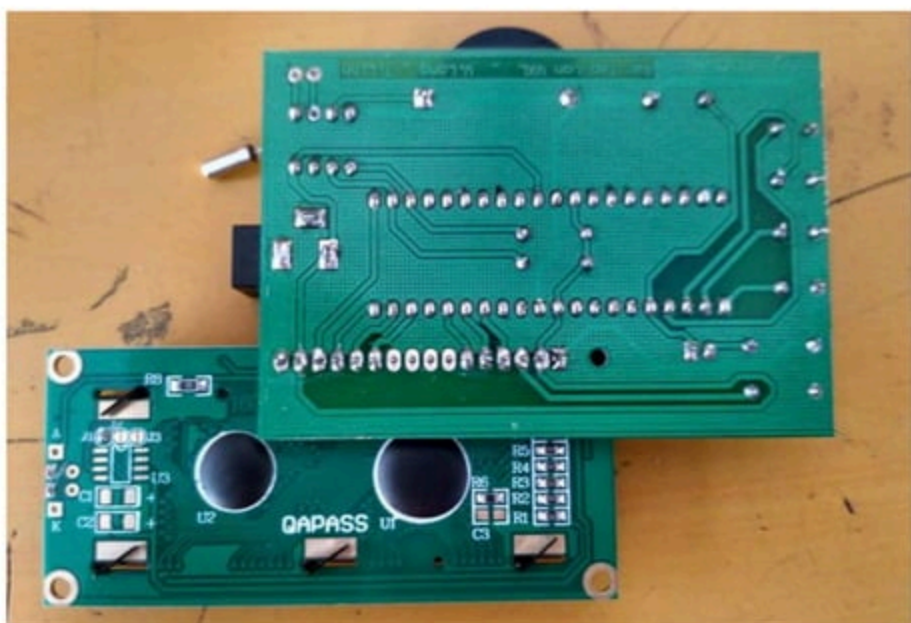
Hình 18. Kiểm tra mạch trên board



Hình 19. Mạch in chưa hàn linh kiện



Hình 20. Hình ảnh mạch thực tế đang hoạt động



Hình 21. Mặt sau mạch hoàn chỉnh

3. Đo đạc, kiểm tra, đánh giá chất lượng sản phẩm

Việc đo nguội là quá trình đo đạc, kiểm tra trước khi cấp nguồn cho mạch. Việc này nhằm kiểm tra kết nối giữa cách đường đi dây, các kết nối trên mạch in, các mối hàn.

Sau khi đo nguội mà không có lỗi, ta sẽ cấp nguồn cho mạch hoạt động (Chú ý : cấp nguồn sau cùng khi lắp ráp mạch và ngắt nguồn đầu tiên khi tắt mạch).

Kiểm tra các chức năng hoạt động của đồng hồ, LCD hiển thị, chức năng các phím nhấn, led báo hiệu, còi báo thức có hoạt động tốt không.

Khi mạch đã hoạt động ổn định, ta cần đo sai số của mạch. Nếu mạch có sai số lớn hơn mức cho phép, cần điều chỉnh lại code. Chúng em đã kiểm tra và đo được sai số trong 1 ngày hoạt động là 35 giây.

Trong quá trình hoạt động, cần đánh giá nhiệt độ của mạch để có biện pháp điều chỉnh giúp mạch tản nhiệt tốt, bảo vệ linh kiện và hoạt động ổn định hơn.

Chương VI. Kết luận

1. Đánh giá những điều đã làm được và chưa làm được

1.1 Những điều đã làm được

Nhìn chung, sản phẩm hoàn thành đã đạt được yêu cầu thiết kế đã đề ra:

- Đồng hồ Lịch vạn niên đếm thời gian một cách chính xác. Có các chức năng chỉnh giờ - phút, thứ - ngày - tháng - năm và cài đặt báo thức.
- Sản phẩm đơn giản, gọn nhẹ, dễ dàng sử dụng. Sản phẩm sử dụng dòng điện dân dụng và thích hợp với việc đặt cố định một chỗ.

Ngoài việc áp dụng những kiến thức đã được học trong môn vi xử lý, chúng em còn được làm quen và thực hành với Vi điều khiển AVR Atmega16. Đây là những kiến thức nền tảng, phục vụ cho ngành học điện tử sau này.

Chúng em được làm quen và sử dụng các phần mềm mô phỏng Proteus, phần mềm lập trình CodeVisionAVR C Compiler cho vi điều khiển AVR, phần mềm thiết kế mạch in Altium và thực hiện ráp mạch thực tế. Đây là những kỹ năng rất quan trọng trong quá trình học tập, đặc biệt đối với sinh viên ngành điện tử.

Trong quá trình hoàn thành thiết kế sản phẩm Mạch đồng hồ Lịch vạn niên sử dụng vi xử lý AVR, chúng em đã học hỏi được thêm rất nhiều kiến thức và kinh nghiệm về điện tử. Chúng em được thực hành, vận dụng các kiến thức đã học để thiết kế nên một sản phẩm theo đúng quy trình thiết kế sản phẩm điện tử chuyên nghiệp.

1.2 Những điểm còn hạn chế

Ngoài các chức năng đã đạt được đề tài còn một số hạn chế sau:

- Mạch nhìn còn thô, tính thẩm mỹ chưa cao.
- Các nút bấm còn chưa nhạy, chưa có nút reset mạch (phải reset mạch bằng cách rút nguồn).

- Mạch chạy thời gian dài còn tỏa nhiều nhiệt, ảnh hưởng tới độ bền của linh kiện và độ chính xác của sản phẩm.

2. Hướng phát triển

Đề tài có thể được tiếp tục phát triển một số chức năng sau:

- Thêm phần hiển thị lịch âm.
- Thêm phần đo nhiệt độ, độ ẩm của môi trường.
- Hệ thống chuông báo là bài hát/đoạn âm thanh.

3. Kết luận

Bài tập lớn Vi xử lý đã giúp chúng em hoàn thiện hơn kỹ năng làm việc nhóm, vận dụng các kiến thức đã được học để thiết kế ra một sản phẩm điện tử hoàn chỉnh. Đề tài Mạch đồng hồ Lịch vạn niên đã cho em nhiều trải nghiệm bổ ích, mở ra nhiều hướng đi mới cho học tập và công việc sau này.

Chúng em xin được gửi lời cảm ơn chân thành tới thầy TS. Nguyễn Hoàng Dũng, anh Thành và các anh trợ giảng đã giúp đỡ chúng em rất nhiều trong các quá trình thiết kế và hoàn thành sản phẩm.

Tài liệu tham khảo

- [1] ThS. Phạm Hùng Kim Khánh, *Giáo trình Vi Điều Khiển*
- [2] ThS. Trần Nhật Khải Hoàn, GS.Trần Hữu Danh, *Tài liệu hướng dẫn thực tập Vi Điều Khiển.*
- [3] ThS. Lương Văn Sơn, *Giáo trình mạch Xung.*
- [4] ThS. Nguyễn Trung Tập, *Giáo trình mạch Số*
- [5] GS. Trương Văn Tám, *Giáo trình mạch Tương Tự.*
- [6] <http://www.alldatasheet.com>
- [7] <http://www.hocavr.com>
- [8] <http://www.dientuvietnam.net>
- [9] <http://codientu.org>

Phụ lục 1: Source code

```

/*****
This program was produced by the
CodeWizardAVR V2.05.0 Professional
Automatic Program Generator
© Copyright 1998-2010 Pavel Haiduc, HP InfoTech s.r.l.
http://www.hpinfotech.com

Project : Dong ho lich van nien su dung VDK ATmega16
Date    : 15-May-2016
Author  : HoangTuanLinh & LongNguyenVan
Company : Hanoi University of Science and Technology
Chip type           : ATmega16
Program type        : Application
AVR Core Clock frequency: 8.000000 MHz
Memory model        : Small
External RAM size    : 0
Data Stack size     : 256
*****/

#include <mega16.h>
#include <delay.h>

// I2C Bus functions
#define
    .equ __i2c_port=0x15 ;PORTC
    .equ __sda_bit=1
    .equ __scl_bit=0
#define
#include <i2c.h>

// DS1307 Real Time Clock functions
#include <ds1307.h>

// Alphanumeric LCD Module functions
#include <alcd.h>

// Declare your global variables here
#define MODE    PINB.0

```



```

#define UP      PINB.1
#define DOWN    PINB.2
#define OK      PINB.3

bit AP; // che do 24h (1), che do 12h(0)
bit sb; // sb=1: phim duoc nhan; sb=0; phim khong dc nhan
bit alarm_en;
unsigned int h_set=0,m_set=0;
unsigned char hour,min,sec,day,date,month,year,mode,h,No_date;

//Chuong trinh con
//Ham hien thi thu (day)
void display_day(unsigned char x){
    switch(x) {
        case 1: lcd_putsf(" 2");
            break;
        case 2: lcd_putsf(" 3");
            break;
        case 3: lcd_putsf(" 4");
            break;
        case 4: lcd_putsf(" 5");
            break;
        case 5: lcd_putsf(" 6");
            break;
        case 6: lcd_putsf(" 7");
            break;
        case 7: lcd_putsf(" 8");
            break;
        default:
            };
    }

//Ham doi ngay
void check_conveter_day(void){
    if(day ==8) day=1;
    if (day == 0) day = 7;
    rtc_set_date(day, date,month,year);
}

//Ham doi che do 12h <-> 24h
void check_conveter_hour(void){
    //kiem tra che do gio hien tai

```



```

    if (AP == 0) { //che do 12h (AP = 0)
        if (hour > 12) h = hour-12;
        else h = hour;
    }
    else h = hour; //che do 24h (AP = 1)
}

//Ham hien thi time len LCD
void display_time_lcd(unsigned char x,unsigned char y){
    lcd_gotoxy(x,y);
    lcd_putchar(48+h/10); //hang chuc
    lcd_putchar(48+h%10); //hang don vi
    lcd_putsf(":");
    lcd_putchar(48+min/10);
    lcd_putchar(48+min%10);
    lcd_putsf(":");
    lcd_putchar(48+sec/10);
    lcd_putchar(48+sec%10);
    if (alarm_en == 1) lcd_putsf(" "); //co bao thuc
    else lcd_putsf(" ");
}

//Hien thi date len LCD
void display_date_lcd(unsigned char x,unsigned char y)
{
    lcd_gotoxy(x,y);
    lcd_putchar(48+date/10); //hang chuc
    lcd_putchar(48+date%10); //hang don vi
    lcd_putsf("/");
    lcd_putchar(48+month/10);
    lcd_putchar(48+month%10);
    lcd_putsf("/");
    lcd_putchar(48+year/10);
    lcd_putchar(48+year%10);
    display_day(day); //hien thi thu
}

//Ham hien thi o che do hen gio
void display_hengio_lcd(unsigned char x,unsigned char y)
{
    lcd_gotoxy(x,y);

```

```

    lcd_putchar(48+h_set/10);
    lcd_putchar(48+h_set%10);
    lcd_putsf(":");
    lcd_putchar(48+m_set/10);
    lcd_putchar(48+m_set%10);
    if (alarm_en == 1) lcd_putsf(" *"); //có báo thức
        else lcd_putsf(" ");
}

//Ham hien thi len LCD
void display_lcd(void){
    // Hien thi time len lcd
    lcd_gotoxy(0,0);
    lcd_putsf("Time: "); //Hien thi thoi gian
    display_time_lcd(6,0);

    // Hien thi date len lcd
    lcd_gotoxy(0,1);
    lcd_putsf("Date: ");
    display_date_lcd(6,1);
    //display_msg_lcd(); // chay msg tren LCD
}

//Chuong trinh con kiem tra trang thai cua phim nhan : bit trang thai
sb
void status_button(void) {
    if(MODE&&UP&&DOWN&&OK&&sb) {
        sb=0; // kiem tra phim da nha hay chua?
        //PORTB.1 = sb;
    }
}

//Ham kiem tra so ngay trong thang
void Ndate_in_month(void){
    if (month == 2){
        if (year%4 == 0) No_date = 29;
        else No_date = 28;
    }

    else if (month ==1 || month ==3 || month ==5 || month ==7 ||
month == 8 || month == 10 || month ==12) No_date = 31;
        else No_date = 30;
}

```

```

}

//Ham cai dat thoi gian & bao thuc
void set_alarm_time (void){
    status_button(); //Kiem tra trang thai phim bam da nha hay chua

    //chuyen doi che do 12h <-> 24h
    if ((OK == 0) && (sb == 0)){ //phim SET duoc nhan khi cac phim khac
        khong duoc nhan
        AP = ~AP; //doi che do gio hien thi
        sb = 1; //bao co phim nhan
        //PORTB.1 = sb;
    }
    status_button();
    //chon che do nguoi dung
    if (MODE==0 && sb==0){ //chon che do
        if (mode == 9){
            //lcd_clear();
            mode = 0;
        }
        else {
            //lcd_clear();
            mode++;
        }
        sb = 1; //bao co phim nhan
        //PORTB.1 = sb;
    }

    //che do mode = 0 (hien thi thong tin de tai)
    if (mode == 0) {
        lcd_gotoxy(0,0);
        lcd_putsf("BAI TAP LON VXL");
        lcd_gotoxy(0,1);
        lcd_putsf("V.LONG - T.LINH");
        delay_ms (1000); //sau 1s tu chuyen sang che do hien thi binh
        thuong
        mode++;
    }

    //che do mode = 1 (hien thi binh thuong)
    if(mode==1){

```

```

display_lcd();
if((alarm_en == 1) && (hour == h_set) && (min == m_set)) {
    PORTB.4 = 1; //bao thuc keu
    status_button();
    if (OK == 0 && sb == 0){ //bao thuc
        alarm_en = 0; //tat bao thuc
        sb = 1;
        //PORTB.1 = sb;
    }
}
else {
    PORTB.4 = 0; // tat bao thuc
    //alarm_en = 0; //tu dong tat bao thuc sau 1 phut
}
}

//che do mode=2 (chinh gio)
if (mode == 2){ //che do chinh gio
    //lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("Set Hour: ");
    lcd_putsf(" ");
    display_time_lcd(5, 1);
    lcd_putsf(" ");
    status_button();
    if (UP == 0 && sb == 0){ //Phim UP duoc nhan
        if (hour == 23) hour = 0;
        else hour++;
        rtc_set_time (hour, min, sec); //dat lai gio cho DS1307
        sb = 1;
        //PORTB.1 = sb;
    }
    status_button();
    if (DOWN == 0 && sb == 0){
        if (hour == 0) hour = 23;
        else hour--;
        rtc_set_time(hour, min, sec);
        sb = 1;
        //PORTB.1 = sb;
    }
}
}

```

```

//che do mode = 3 (chỉnh phút)
if (mode == 3){ //che do chỉnh phút
    //lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("Set Minute: ");
    lcd_putsf("      ");
    display_time_lcd(5, 1);
    lcd_putsf(" ");
    status_button();
    if (UP == 0 && sb == 0){
        if (min == 59) min = 0;
        else min++;
        rtc_set_time(hour, min, sec);
        sb = 1;
        //PORTB.1 = sb;
    }
    status_button();
    if (DOWN == 0 && sb == 0){
        if (min == 0) min = 59;
        else min--;
        rtc_set_time(hour, min, sec);
        sb = 1;
        //PORTB.1 = sb;
    }
}

//che do mode = 4 (chỉnh thu(day) đồng thời chỉnh ngày(date))
if (mode == 4){
    //lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("Set Date: ");
    lcd_putsf("      ");
    display_date_lcd(6, 1);
    lcd_putsf(" ");
    Ndate_in_month();
    status_button();
    if (UP == 0 && sb == 0){
        if (day == 7) day = 1;
        else day++;
        //xác định ngày (date) khi chỉnh thu
    }
}

```

```

        if (date == No_date) {
            if (month == 12){ //Chuyen sang nam moi
                date = 1; month = 1; year++;
            }
            else{
                date = 1; month++; //Het thang thi nhay sang
thang moi
            }
        }
        else date++;
        rtc_set_date(day,date,month,year);
        sb=1;
        //PORTB.1 = sb;
    }
    status_button();
    if (DOWN == 0 && sb == 0){
        if (day == 1) day = 7;
        else day--;
        if (date == 1) { //VD: tu 1/3 chuyen xuong 28(hoac 29)/2
            if (month == 1) {
                month = 12; date = 31; year--;
            }
            else{
                month--;
                Ndate_in_month();
                date = No_date;
            }
        }
        else date--;
        rtc_set_date(day, date,month,year);
        sb=1;
        //PORTB.1 = sb;
    }
}

//che do mode = 5 (chinh thang)
if (mode == 5){
    lcd_gotoxy(0,0);
    lcd_putsf("Set Month: ");
    lcd_putsf("          ");
    display_date_lcd(6, 1);
}

```

```

lcd_putsf(" ");
Ndate_in_month();
status_button();
if (UP == 0 && sb == 0){ //phím UP được nhấn
    if (month == 12) {
        month = 1;
        if (year==99) year=0;
        else year++;
    }
    else month++;
    //xác định thu (day) khi chỉnh tháng (month)
    if ( (day + No_date%7) > 7 )
        day = day + No_date%7 - 7;
    else day = day + No_date%7;
    Ndate_in_month(); //Cập nhật số ngày tối đa của tháng vừa đặt
    lại
    if (date > No_date) date = No_date; //VD: từ ngày 31/3 phải
    chuyển thành 30/4 khi ấn UP(month)
    rtc_set_date(day, date, month, year);
    sb = 1;
    //PORTB.1 = sb;
}
status_button();
if (DOWN == 0 && sb == 0){
    if (month == 1) {
        month = 12;
        if (year == 0) year=99;
        else year--;
    }
    else month--;
    Ndate_in_month(); //Kiểm tra, phải tính theo số ngày của tháng
    trước để xác định thu
    if (day < No_date%7) day = day - No_date%7 +7;
    else day = day - No_date%7;
    if (date > No_date) date = No_date; //VD: từ ngày 31/3 phải
    chuyển thành 29/2 khi ấn DOWN(month)
    rtc_set_date(day, date, month, year);
    sb = 1;
    //PORTB.1 = sb;
}
}

```



```

//che do moode = 6 (chinh nam)
if (mode == 6){
    lcd_gotoxy(0,0);
    lcd_putsf("Set Year: ");
    lcd_putsf("          ");
    display_date_lcd(6, 1);
    //lcd_putsf(" ");
    Ndate_in_month();
    status_button();
    if (UP == 0 && sb == 0){ //phim UP duoc nhan
        //Chinh ngay cua thang 2 khi chuyen tu nam nhuan sang nam khong
nhuan
        if ((year%4==0) && (month==2) && (date==29)){ //ngay 29/2 nam
nhuan
            date = 28;
        }
        //Chinh thu(day) khi chuyen nam(year)
        if (year%4 == 0){ //nam nhuan
            if ((month == 1) || ((month==2)&&(date<29))){
                if (day == 7) //thu CN
                    day = 2; //thu 3
                else day = day + 2;
            }
            else day = day + 1; //nam nhuan nhưng thu chi tang 1
ngay
        }
        else day = day + 1; //khong phai nam nhuan
        //Chuyen nam
        if (year == 99)
            year = 0;
        else
            year++;
        rtc_set_date(day, date, month, year);
        sb = 1;
        //PORTB.1 = sb;
    }
    status_button();
    if (DOWN == 0 && sb == 0){
        //Chuyen nam
        if (year%4 == 0) { //nam nhuan

```

```

        if ((month > 2) || ((month==2) && (date==29))) {
            if (day == 1) //thu 2
                day = 6; //thu 7
            else day -= 2;
        }
        else day--1; //nam nhuan nhưng thu chỉ lùi 1 ngày
    }

    else day -= 1; //không phải năm nhuận
    //Chỉnh ngày của tháng 2 khi chuyển từ năm nhuận sang năm không
nhuan
    if ((year%4==0) && (month==2) && (date==29)) { //ngày 29/2 năm
nhuan
        date = 28;
    }
    //Chỉnh năm
    if (year == 0) year = 99;
    else year--;
    rtc_set_date(day, date, month, year);
    sb = 1;
    //PORTB.1 = sb;
}

//che do mode = 7 ({chỉnh giờ trong chế độ hẹn giờ})
if (mode == 7) {
    lcd_gotoxy(0,0);
    lcd_putsf("Alarm: ");
    display_hengio_lcd(8,0);
    lcd_gotoxy(0,1);
    lcd_putsf("Hour: ");
    lcd_gotoxy(8,1);
    lcd_putchar(h_set/10 + 48);
    lcd_putchar(h_set%10 + 48);
    lcd_putsf("      ");
    status_button();
    if (UP == 0 && sb == 0) {
        if (h_set == 23) h_set = 0;
        else h_set++;
        sb = 1;
        //PORTB.1 = sb;
    }
    //status_button();
}

```

```

        if (DOWN == 0 && sb == 0) {
            if (h_set == 0) h_set = 23;
            else h_set--;
            sb = 1;
            //PORTB.1 = sb;
        }
        //status_button();
        if (OK == 0 && sb == 0) {
            alarm_en = ~alarm_en; //xác nhận cài đặt xong báo thức;
            alarm_en = 1 (0) -> có (không có) báo thức
            //lcd_gotoxy(15,0);
            //lcd_putsf("*");
            sb = 1;
        }
    }

    //che do mode = 8 (chỉnh phút trong chế độ hẹn giờ)
    if (mode == 8) {
        lcd_gotoxy(0,0);
        lcd_putsf("Alarm: ");
        display_hengio_lcd(8,0);
        lcd_gotoxy(0,1);
        lcd_putsf("Minute: ");
        lcd_gotoxy(8,1);
        lcd_putchar(m_set/10 + 48);
        lcd_putchar(m_set%10 + 48);
        lcd_putsf("      ");
        status_button();
        if (UP == 0 && sb == 0) {
            if (m_set == 59) m_set = 0;
            else m_set++;
            sb = 1;
            //PORTB.1 = sb;
        }
        //status_button();
        if (DOWN == 0 && sb == 0) {
            if (m_set == 0) m_set = 59;
            else m_set--;
            sb = 1;
            //PORTB.1 = sb;
        }
        //status_button();
    }

```

```

        if(OK == 0 && sb == 0){
            alarm_en = ~alarm_en;    //xác nhận cài đặt xong báo thức;
            alarm_en = 1 (0) -> có (không có) báo thức
            sb = 1;
        }
        //PORTB.1 = sb;
    }
}

//_____Ham chinh_____
void main(void)
{
    // Declare your local variables here

    // Input/Output Ports initialization
    // Port A initialization
    // Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
    Func0=In
    // State7=T State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTA=0x00;
    DDRA=0x00;

    // Port B initialization
    // Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out
    Func1=Out Func0=Out
    // State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0
    State0=0
    PORTB=0xEF; //1111_1111
    DDRB=0x10;  //1111_0000

    // Port C initialization
    // Func7=Out Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In
    Func0=In
    // State7=0 State6=T State5=T State4=T State3=T State2=T State1=T
    State0=T
    PORTC=0xFF;
    DDRC=0x80; //1000_0000

    // Port D initialization

```

```
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out
Func1=Out Func0=Out
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0
State0=0
PORTD=0x00;
DDRD=0xFF;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=0xFF
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=0xFFFF
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
```

```
// Clock value: Timer2 Stopped
// Mode: Normal top=0xFF
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// USART disabled
UCSRB=0x00;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// ADC initialization
// ADC disabled
ADCSRA=0x00;

// SPI initialization
// SPI disabled
SPCR=0x00;

// TWI initialization
// TWI disabled
TWCR=0x00;

// I2C Bus initialization
```



```

i2c_init();

// DS1307 Real Time Clock initialization
// Square wave output on pin SQW/OUT: Off
// SQW/OUT pin state: 0
rtc_init(0,0,0);
// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTD Bit 0
// RD - PORTD Bit 1
// EN - PORTD Bit 2
// D4 - PORTD Bit 4
// D5 - PORTD Bit 5
// D6 - PORTD Bit 6
// D7 - PORTD Bit 7
// Characters/line: 16
lcd_init(16);

//cai dat mac dinh
rtc_set_time(0, 0, 0);
rtc_set_date(1, 16, 5, 16);
AP = 1; //mac dinh de che do la 24h
sb = 0; //mac dinh khong co nut bam
//alarm_en = 0;
//PORTB.4 = 0;
while (1)
{
    // Place your code here
    rtc_get_time(&hour,&min,&sec);    // Doc gio, phut, giay tu
    ds1307
    rtc_get_date(&day, &date,&month,&year);    // Doc thu, ngay,
    thang, nam tu ds1307
    check_conveter_day();    // kiem tra va kiem chuyen thu, che
    do gio ve dang chuan
    check_conveter_hour();
    status_button();
    set_alarm_time(); // goi ctc thuc hien cai dat
    //PORTB.0 = 0;
}
}

```

Phụ lục 2: Hướng dẫn sử dụng

Mạch đồng hồ Lịch vạn niên có các chức năng sau:

- Hiển thị thời gian : giờ, phút, giây.
- Hiển thị thứ, ngày, tháng, năm.
- Có thể chỉnh sửa thông số thời gian, ngày tháng năm.
- Có chức năng đặt giờ báo thức, sử dụng đèn Led và còi báo hiệu.

Sử dụng các phím MODE để vào chế độ Cài đặt và chỉnh sửa:

- Cài đặt giờ : ấn phím UP hoặc DOWN để thay đổi
- Cài đặt phút : ấn phím UP hoặc DOWN để thay đổi
- Cài đặt ngày : ấn phím UP hoặc DOWN để thay đổi
- Cài đặt tháng : ấn phím UP hoặc DOWN để thay đổi
- Cài đặt năm : ấn phím UP hoặc DOWN để thay đổi.
- Chế độ báo thức :
 - Đặt giờ : ấn phím UP hoặc DOWN để thay đổi.
 - Đặt phút : ấn phím UP hoặc DOWN để thay đổi.
 - Xác nhận đặt báo thức bằng phím OK. Khi đó sẽ có kí hiệu * để báo hiệu trên màn hình.

Khi đến thời điểm đặt báo thức, sẽ có còi và Led báo hiệu. Người sử dụng sẽ ấn phím OK nếu muốn tắt thông báo trên. Nếu không tắt báo thức thì đèn và còi sẽ tự động hủy báo thức sau 1 phút, trong trường hợp này báo thức vẫn sẽ hoạt động vào ngày hôm sau.