

**TRƯỜNG ĐẠI HỌC PHENIKAA**

**KHOA: CÔNG NGHỆ THÔNG TIN**

**A logo for a university

Description automatically generated**

**ĐỒ ÁN HỆ NHÚNG**

**ĐỀ TÀI:** **THIẾT KẾ HỆ THỐNG MỞ KHÓA CỬA THÔNG MINH & ĐIỀU KHIỂN CÁC THIẾT BỊ TRONG NHÀ**

**GVDH: Vũ Văn Quang**

**SVTH: Nhóm 4:**

**Trần Duy Nam - 21012079**

**Vũ Ngọc Văn - 21012100**

**Phan Thúy Nga - 21012080**

**Lớp: CSE703018-1-2-23(N01)**

**Hà Nội, Tháng 3 Năm 2024**

# LỜI NÓI ĐẦU

**-----****-----**

Trong thời kỳ ngày nay, khi sự tiến bộ khoa học và kỹ thuật đang ngày càng phát triển, thế giới chúng ta đang chứng kiến một cuộc cách mạng về mặt văn minh và hiện đại. Cụ thể, sự tiến bộ đáng kể trong lĩnh vực kỹ thuật điện tử đã mở ra những khả năng mới, tạo ra những thiết bị thông minh với những đặc tính nổi bật như độ chính xác cao, tốc độ nhanh, và tính di động, tất cả đều mang lại những lợi ích quan trọng cho cuộc sống hàng ngày.

Cụ thể, ứng dụng của kỹ thuật điện tử không chỉ giúp tự động hóa nhiều khía cạnh trong cuộc sống hàng ngày của con người mà còn thúc đẩy sự phát triển của các hệ thống thông minh. Các hệ thống nhà cửa, nhà máy, trường học, giao thông,… Những hệ thống này không chỉ mang lại tính an toàn cao mà còn tạo ra môi trường sống thông minh, cho phép người dùng kiểm soát và quản lý các thiết bị trong môi trường sinh hoạt một cách linh hoạt và thuận tiện. Đây là một hướng nghiên cứu mang tính đổi mới, đáp ứng nhu cầu ngày càng cao của xã hội đương đại.

Xuất phát từ ứng dụng trên, nhóm em đã chọn đề tài “Thiết kế hệ thống mở khóa cửa

thông minh& điều khiển các thiết bị trong nhà” cho môn Hệ nhúng.

# LỜI CẢM ƠN

**-----****-----**

Nhóm em muốn bày tỏ lòng biết ơn chân thành đến thầy Vũ Văn Quang đã dành thời gian và công sức hướng dẫn tận tình, cũng như tạo điều kiện thuận lợi cho nhóm em trong quá trình thực hiện đề tài. Sự nhiệt huyết và sự hỗ trợ không ngừng từ thầy đã đóng góp quan trọng vào việc xây dựng nền tảng kiến thức của nhóm em, giúp em vượt qua những thách thức trong quá trình nghiên cứu.

Trong suốt hành trình này, em nhận thức được rằng không tránh khỏi những thiếu sót và khuyết điểm. Em trân trọng những góp ý và sự chỉ dẫn từ quý thầy/cô, và em cam kết sẽ tích cực thấu hiểu và áp dụng những phản hồi quý báu đó để đưa đề tài của mình lên một tầm cao mới. Sự hỗ trợ và tình cảm quý báu của quý thầy/cô là động viên lớn, là động lực mạnh mẽ giúp em không ngừng phát triển trong sự nghiệp nghiên cứu. Xin chân thành cảm ơn!Em xin chân thành cảm ơn!

# NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

….......................................................................................................................................

...........................................................................................................................................

................................................................................................................................….......

................................................................................................................................….......

................................................................................................................................….......

................................................................................................................................….......

................................................................................................................................….......

................................................................................................................................….......

................................................................................................................................….......

...............................................................................................................................…........

...............................................................................................................................…........

...............................................................................................................................…........

...........................................................................................................................................

….......................................................................................................................................

….......................................................................................................................................

….......................................................................................................................................

Hà Nội, ngày … tháng … năm 2024

## GIẢNG VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

# TIẾN ĐỘ THỰC HIỆN ĐỒ ÁN MÔN HỌC

**Tên đề tài:** THIẾT KẾ HỆ THỐNG MỞ KHÓA CỬA THÔNG MINH & ĐIỀU KHIỂN CÁC THIẾT BỊ TRONG NHÀ

**Ngày giao đề tài:** //; **Tuần thứ: 1**

## Ngày hoàn thành đề tài: 4/3/2024; Tuần thứ: Nhóm sinh viên thực hiện: Nhóm 4

Họ tên sinh viên:

Trần Duy Nam…..MSSV: 21012079

Vũ Ngọc Văn……MSSV: 21012100

Phan Thúy Nga….MSSV: 21012080

|  |  |
| --- | --- |
| ***Tuần/ngày*** | ***Nội dung – công việc thực hiện*** |
| Tuần 1,2,3 | Đăng kí đồ án, gặp giáo viên hướng dẫn để đề xuất đề tài đồ án,  chờ giáo viên hướng dẫn xét duyệt đề tài đồ án. |
| Tuần 4,5,6 | Tìm hiểu lên ý tưởng và thiết kế sơ đồ nguyên lí |
| Tuần 7,8,9 | Kiểm tra sơ đồ nguyên lý, sửa lỗi, mô phỏng và giải thích sơ đồ  nguyên lý của mạch |
| Tuần 10,11,12 | Thi công mạch, chỉnh sửa lỗi và lắp ráp mạch, kiểm tra mạch thực  tế chạy hay không để có hướng sửa chữa. |
| Tuần 13,14,15 | Tiến hành viết báo cáo và đưa cho giáo viên hướng dẫn xem và  chỉnh sửa. |
| Tuần 16 | Gửi báo cáo cho giáo viên hướng dẫn kiểm tra lần cuối xem xét và  góp ý kiến, sau đó nộp đồ án và báo cáo trước khi bảo vệ. |

Xác nhận của giáo viên hướng dẫn (Ký và ghi rõ họ và tên)

# LỜI CAM ĐOAN

**-----****-----**

Đề tài này là do nhóm em tự thực hiện dựa vào một số tài liệu và không sao chép từ tài liệu hay công trình đã có trước đó. Nếu có sao chép em xin hoàn toàn chịu trách nhiệm.

Hà Nội, ngày 04 tháng 03 năm 2024

## NGƯỜI VIẾT

**NHÓM 4**

# MỤC LỤC

**-----****-----**

[LỜI NÓI ĐẦU 1](#_bookmark0)

[LỜI CẢM ƠN 3](#_bookmark1)

[NHẬN XÉT CỦA GIÁO VIÊN HƢỚNG DẪN 4](#_bookmark2)

[TIẾN ĐỘ THỰC HIỆN ĐỒ ÁN MÔN HỌC 5](#_bookmark3)

[LỜI CAM ĐOAN 6](#_bookmark4)

[DANH MỤC HÌNH ẢNH 9](#_bookmark5)

[DANH MỤC BẢNG 10](#_bookmark6)

[CHƢƠNG I: TỔNG QUAN VỀ ĐỀ TÀI 11](#_bookmark7)

* 1. [Giới thiệu đề tài 11](#_bookmark8)
  2. [Phƣơng tiện nghiên cứu. 11](#_bookmark9)
  3. [Mục đích và yêu cầu đề tài. 11](#_bookmark10)

[CHƢƠNG II: CƠ SỞ THU ẾT 12](#_bookmark11)

* 1. [IC thời gian thực DS1307 12](#_bookmark12)
  2. [Hệ nhúng ATMEGA16 13](#_bookmark14)

[2.3 LCD 16x2 16](#_bookmark16)

* 1. [Chuông điện 17](#_bookmark18)
  2. [Điện trở 18](#_bookmark20)
  3. [Relay 19](#_bookmark22)

[CHƢƠNG III: THIẾT KẾ VÀ THI CÔNG MẠCH 20](#_bookmark24)

* 1. [Sơ đồ khối. 20](#_bookmark25)
  2. [Sơ đồ nguyên lý và chức năng của từng khối. 20](#_bookmark27)
     1. [Khối công suất 20](#_bookmark28)
     2. [Khối hệ nhúng(Atmega16) 21](#_bookmark30)
     3. [Khối thời gian thực 22](#_bookmark32)
     4. [Khối hiển thị LCD 16X2 23](#_bookmark34)
  3. [Thi công. 25](#_bookmark37)
     1. [Mạch in 25](#_bookmark38)
     2. [Mạch hoàn chỉnh 26](#_bookmark40)
  4. [Code chƣơng trình hệ nhúng: 27](#_bookmark42)

[CHƢƠNG IV: KẾT QUẢ THỰC HIỆN VÀ KẾT LUẬN 36](#_bookmark43)

* 1. [Ƣu , nhƣợc điểm của sản phẩm 36](#_bookmark44)
  2. [Kết luận. 36](#_bookmark45)

[TÀI LIỆU THAM KHẢO 37](#_bookmark46)

# DANH MỤC HÌNH ẢNH

**-----****-----**

[Hình 2. : IC Thời gian thực DS1307 12](#_bookmark13)

[Hình 2.2: Hệ nhúng ATMEGA16 13](#_bookmark15)

[Hình 2.3: LCD 16x2 16](#_bookmark17)

[Hình 2.4: Chuông điện 220V 17](#_bookmark19)

[Hình 2.5: Điện trở dạng cắm 18](#_bookmark21)

[Hình 2.6: Relay 19](#_bookmark23)

[Hình 3. : Sơ đồ khối 20](#_bookmark26)

[Hình 3.2: Khối công suất 20](#_bookmark29)

[Hình 3.3: Khối hệ nhúng 21](#_bookmark31)

[Hình 3.4: Khối thời gian thực 22](#_bookmark33)

[Hình 3.5: Khối hiển thị 23](#_bookmark35)

[Hình 3.6: Sơ đồ nguyên lý chung 24](#_bookmark36)

[Hình 3.7: Sơ đồ mạch in 25](#_bookmark39)

[Hình 3.8: Hình ảnh mạch hoàn chỉnh 26](#_bookmark41)

# DANH MỤC BẢNG

Bảng 1.1: Thời gian chuông kêu trong ngày… 11

Bảng 2.1: Chức năng từng chân của IC DS 307… 13

Bảng 2.2: Chức năng từng chân của LCD 6x2 6

# CHƯƠNG I: TỔNG QUAN VỀ ĐỀ TÀI

## Giới thiệu đề tài

* + - Với sự bùng nổ của Internet of Things (IoT), khái niệm ngôi nhà thông minh không chỉ đơn thuần là một xu hướng mà còn là nhu cầu thiết yếu. Ngôi nhà thông minh không chỉ là nơi ẩn mình của gia đình mà còn là trung tâm điều khiển linh hoạt cho mọi hoạt động.
    - Để giải quyết thách thức quản lý an ninh và điều khiển thiết bị, em quyết định tập trung nghiên cứu và phát triển "Hệ Thống Mở Khóa Cửa Thông Minh & Điều Khiển Các Thiết Bị Trong Nhà".

## Phương tiện nghiên cứu.

* Tài liệu tổng hợp từ một số nguồn khác nhau.
* Tìm kiếm thông tin trên internet.
* Các phần mềm hỗ trợ: Protues, Arduino IDE

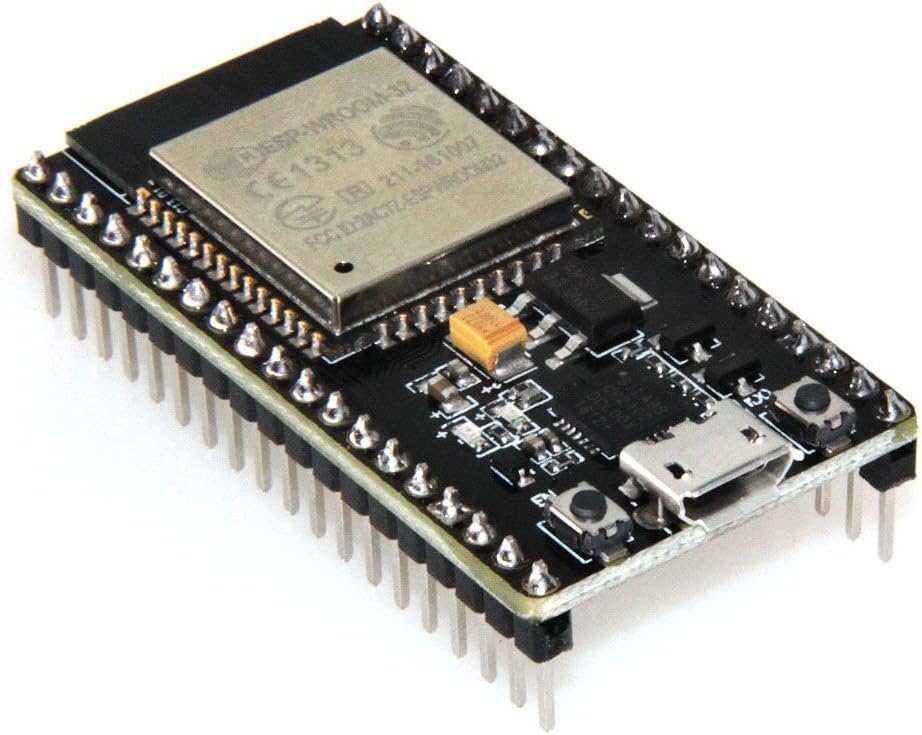
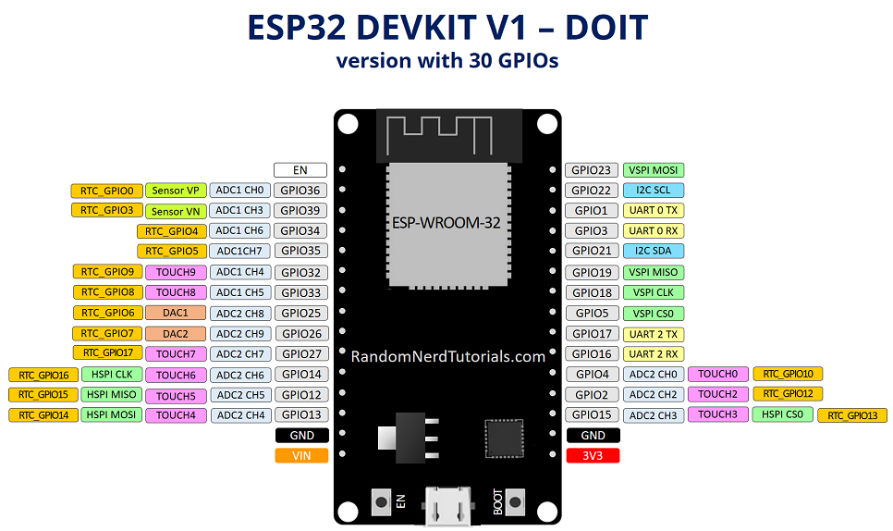
## Mục đích và yêu cầu đề tài.

* Hệ thống mở cửa từ xa, quản lý và thay đổi mật khẩu cửa
* Có khả năng ghi lại thời gian ra vào, đóng mở cửa
* Bật tắt các thiết bị trong nhà: Đèn, nóng lạnh, điều hòa
* Hệ thống làm việc ổn định.
* Có thể áp dụng mô hình trong thực tế

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT

Để thực hiện mạch mở khóa cửa thông và điều khiển thiết bị trong nhà cần sử dụng nhiều loại linh kiện khác nhau kết hợp với phần mềm ứng dụng điện thoại di động Flutter để có thể điều khiển và quản lý từ xa một cách linh hoạt, sau đây là một số linh kiện chính có trong mạch và phần mềm ứng dụng điện thoại:

## Modul ESP32

****

### Hình 2.1: Modul Esp32

* + - ESP32 là một bộ vi điều khiển thuộc danh mục vi điều khiển trên chip công suất thấp và tiết kiệm chi phí. Hầu hết tất cả các biến thể ESP32 đều tích hợp Bluetooth và Wi-Fi chế độ kép, làm cho nó có tính linh hoạt cao, mạnh mẽ và đáng tin cậy cho nhiều ứng dụng.
    - Nó là sự kế thừa của vi điều khiển NodeMCU ESP8266 phổ biến và cung cấp hiệu suất và tính năng tốt hơn. Bộ vi điều khiển ESP32 được sản xuất bởi Espressif Systems và được sử dụng rộng rãi trong nhiều ứng dụng khác nhau như IoT, robot và tự động hóa.
    - ESP32 cũng được thiết kế để tiêu thụ điện năng thấp, lý tưởng cho các ứng dụng chạy bằng pin. Nó có hệ thống quản lý năng lượng cho phép nó hoạt động ở chế độ ngủ và chỉ thức dậy khi cần thiết, điều này có thể kéo dài tuổi thọpinrấtnhiều.

### Bảng 2.1: Bảng khả nắng sử dụng từng chân của ESP32

|  |  |  |  |
| --- | --- | --- | --- |
| GPIO Pin | Đầu vào INPUT | Đầu ra  OUTPUT | Chức năng |
| 0 | Pull-up | OK | Xuất tín hiệu PWM khi khởi động |
| 1 | Chân TX (nút) | OK | Đầu ra debug khi khởi động |
| 2 | OK |  | Kết nối đến đèn LED trên bo mạch |
| 3 | OK | RX (nút) | Mức HIGH khi khởi động |
| 4 | OK |  |  |
| 5 | OK |  | Xuất tín hiệu PWM khi khởi động |
| 6 | X | X | Kết nối đến bộ nhớ flash tích hợp |
| 7 | X | X | Kết nối đến bộ nhớ flash tích hợp |
| 8 | X | X | Kết nối đến bộ nhớ flash tích hợp |
| 9 | X | X | Kết nối đến bộ nhớ flash tích hợp |
| 10 | X | X | Kết nối đến bộ nhớ flash tích hợp |
| 11 | X | X | Kết nối đến bộ nhớ flash tích hợp |
| 12 | OK |  | Lỗi khởi động nếu kéo lên cao |
| 13 | OK |  |  |
| 14 | OK |  | Xuất tín hiệu PWM khi khởi động |
| 15 | OK |  | Xuất tín hiệu PWM khi khởi động |
| 16 | OK |  |  |
| 17 | OK | OK |  |
| 18 | OK | OK |  |
| 19 | OK | OK |  |
| 20 | OK | OK |  |
| 21 | OK | OK |  |
| 22 | OK | OK |  |
| 23 | OK | OK |  |
| 24 | OK | OK |  |
| 25 | OK | OK |  |
| 26 | OK | OK |  |
| 27 | OK | OK |  |
| 28 | OK | OK |  |
| 29 | OK | OK |  |
| 30 | OK | OK |  |
| 31 | OK | OK |  |
| 32 | OK | OK |  |
| 33 | OK | OK |  |
| 34 | OK | OK |  |
| 35 | OK | OK |  |
| 36 | OK |  | Chỉ đầu vào |
| 37 | OK |  | Chỉ đầu vào |
| 38 | OK |  | Chỉ đầu vào |
| 39 | OK |  | Chỉ đầu vào |

* + - Các chân được đánh dấu là OK(hỗ trợ) để sử dụng.
    - Các chân được đánh dấu là PULL UP, TX(Chân truyền của UART), RX(Chân nhận của UART)thì có thể sử dụng, nhưng cần chú ý vì chúng có thể có hoạt động không mong muốn chủ yếu khi khởi động.
    - Các chân được đánh dấu X không được khuyến khích sử dụng làm đầu vào hoặc đầu ra .

**Mô tả các chân ESP32:**

**Chân Input Only:** GPIO từ 34 đến 39 là GPI – chân chỉ đầu vào. Các chân này không có điện trở kéo lên hoặc kéo xuống bên trong. Chúng không thể được sử dụng làm đầu ra, vì vậy chỉ sử dụng các chân này làm đầu vào:

* GPIO 34
* GPIO 35
* GPIO 36
* GPIO 39

**Chân tích hợp Flash trên ESP32**: GPIO 6 đến GPIO 11 dùng để kết nối Flash SPI, không khuyến khích sử dụng trong các ứng dụng khác

* GPIO 6 (SCK/CLK)
* GPIO 7 (SDO/SD0)
* GPIO 8 (SDI/SD1)
* GPIO 9 (SHD/SD2)
* GPIO 10 (SWP/SD3)
* GPIO 11 (CSC/CMD)

**Chân cảm biến điện dung:** Các chân ESP32 này có chức năng như 1 nút nhấn cảm ứng, có thể phát hiện sự thay đổi về điện áp cảm ứng trên chân. Các cảm biến cảm ứng bên trong đó được kết nối với các GPIO sau:

* T0 (GPIO 4)
* T1 (GPIO 0)
* T2 (GPIO 2)
* T3 (GPIO 15)
* T4 (GPIO 13)
* T5 (GPIO 12)
* T6 (GPIO 14)
* T7 (GPIO 27)
* T8 (GPIO 33)
* T9 (GPIO 32)

**Analog to Digital Converter (ADC):** ESP32 có các kênh đầu vào ADC 18 x 12 bit (trong khi ESP8266 chỉ có ADC 1x 10 bit). Đây là các GPIO có thể được sử dụng làm ADC và các kênh tương ứng:

* ADC1\_CH0 (GPIO 36)
* ADC1\_CH1 (GPIO 37)
* ADC1\_CH2 (GPIO 38)
* ADC1\_CH3 (GPIO 39)
* ADC1\_CH4 (GPIO 32)
* ADC1\_CH5 (GPIO 33)
* ADC1\_CH6 (GPIO 34)
* ADC1\_CH7 (GPIO 35)
* ADC2\_CH0 (GPIO 4)
* ADC2\_CH1 (GPIO 0)
* ADC2\_CH2 (GPIO 2)
* ADC2\_CH3 (GPIO 15)
* ADC2\_CH4 (GPIO 13)
* ADC2\_CH5 (GPIO 12)
* ADC2\_CH6 (GPIO 14)
* ADC2\_CH7 (GPIO 27)
* ADC2\_CH8 (GPIO 25)
* ADC2\_CH9 (GPIO 26)

**Digital to Analog Converter (DAC):** Có các kênh DAC 2 x 8 bit trên ESP32 để chuyển đổi tín hiệu kỹ thuật số thành đầu ra tín hiệu điện áp tương tự. Các kênh này chỉ có độ phân giải 8 bit, nghĩa là có giá trị từ 0 – 255 tương ứng với 0 – 3.3V. Đây là các kênh DAC

* DAC1 (GPIO25)
* DAC2 (GPIO26)

**Các chân thời gian thực RTC:** Các chân này có tác dụng đánh thức ESP32 khi trong chế độ Low Power Mode. Sử dụng như 1 chân ngắt ngoài.

* RTC\_GPIO0 (GPIO36)
* RTC\_GPIO3 (GPIO39)
* RTC\_GPIO4 (GPIO34)
* RTC\_GPIO5 (GPIO35)
* RTC\_GPIO6 (GPIO25)
* RTC\_GPIO7 (GPIO26)
* RTC\_GPIO8 (GPIO33)
* RTC\_GPIO9 (GPIO32)
* RTC\_GPIO10 (GPIO4)
* RTC\_GPIO11 (GPIO0)
* RTC\_GPIO12 (GPIO2)
* RTC\_GPIO13 (GPIO15)
* RTC\_GPIO14 (GPIO13)
* RTC\_GPIO15 (GPIO12)
* RTC\_GPIO16 (GPIO14)
* RTC\_GPIO17 (GPIO27)

**Chân PWM:** ESP32 LED PWM có 16 kênh độc lập có thể được định cấu hình để tạo tín hiệu PWM với các thuộc tính khác nhau. Tất cả các chân có thể hoạt động như đầu ra đều có thể được sử dụng làm chân PWM (GPIO từ 34 đến 39 không thể tạo PWM). Để xuất PWM, cần xác định các thông số này trong code:

* Frequency – tần số
* Duty cycle
* Kênh PWM
* Chân GPIO nơi bạn muốn xuất tín hiệu

**Chân I2C:** ESP32 có hai kênh I2C và bất kỳ chân nào cũng có thể được đặt làm SDA hoặc SCL. Khi sử dụng ESP32 với Arduino IDE, các chân I2C mặc định là:

* GPIO 21 (SDA)
* GPIO 22 (SCL)

**Chân Ngắt Ngoài:** Tất cả các chân ESP32 đều có thể sử dụng ngắt ngoài

## Arduino Uno R3

|  |  |
| --- | --- |
|  | **Ảnh có chứa Linh kiện điện, mạch điện, Thành phần mạch điện, Kỹ thuật điện  Mô tả được tạo tự động** |

### Hình 2.2: Arduino Uno R3

### Bảng 2.2: Bảng thông số của Arduino Uno R3

|  |  |
| --- | --- |
| Vi điều khiển | ATmega328 họ 8bit |
| Điện áp hoạt động | 5V DC (chỉ được cấp qua cổng USB) |
| Tần số hoạt động | 16 MHz |
| Dòng tiêu thụ | khoảng 30mA |
| Điện áp vào khuyên dung | 7-12V DC |
| Điện áp vào giới hạn | 6-20V DC |
| Số chân Digital I/O | 14 (6 chân hardware PWM) |
| Số chân Analog | 6 (độ phân giải 10bit) |
| Dòng tối đa trên mỗi chân I/O | 30 mA |
| Dòng ra tối đa (5V) | 500 mA |
| Dòng ra tối đa (3V) | 50 mA |
| Bộ nhớ Flash | 32 KB (ATmega38) với 0.5KB dùng bởi bootloader |
| SRAM | 2 KB (ATmega328) |
| EEPROM | 1 KB (ATmega328) |

## LCD 16x2



### Hình 2.3: LCD 16x2

* + - LCD16x2 là một màn hình hiển thị bao gồm nhiều ma trận nhỏ, khi hoạt động thì LCD16x2 sẽ hiển thị các kí tựTrong bảng mã ASCII . Hệ nhúng gửi các tín hiệu khởi tạocho LCD 6X2, sau đó hiển thị các được kí tự lên màn hình hiển thị.
    - Ngày nay, thiết bị hiển thị LCD (Liquid Crystal Display) được sử dụng trong rất nhiều các ứng dụng của VĐK. LCD có rất nhiều ưu điểm so với các dạng hiển thị khác: Nó có khả năng hiển thị kí tự đa dạng, trực quan (chữ, số và kí tự đồ họa), dễ dàng đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau, tốn rất ít tài nguyên hệ thống và giá thành rẻ.

### Bảng 2.3: Chức năng từng chân của LCD 16x2

## Bàn phím ma trận mềm 4x4 Keypad

**Ảnh có chứa biểu đồ, hình vuông, Hình chữ nhật, hàng

Mô tả được tạo tự động**

### Hình 2.4: Bàn phím ma trận mềm 4x4 Keypad

* + - Bàn phím mềm 4×4 keypad là một thiết bị đầu vào thường được sử dụng trong các dự án Arduino để nhập liệu từ người dùng. Đây là một loại bàn phím cảm ứng, thường được thiết kế với ma trận 4 hàng và 4 cột, tạo thành tổng cộng 16 nút nhấn. Mỗi nút nhấn có thể đại diện cho một ký tự, số hoặc chức năng khác.
    - Bàn phím mềm 4×4 keypad có thiết kế nhỏ gọn, dễ kết nối và sử dụng, các chân của 16 phím được nối theo ma trận, tín hiệu khi nhấn phím sẽ là tín hiệu GND (0VDC) hoặc Vcc (5VDC) tùy vào cách quét phím kích vào chân Vi điều khiển, bàn phím còn tích hợp vị trí để lắp thêm tụ chống dội (chống nhiễu), phù hợp cho các ứng dụng điều khiển bằng phím bấm.
    - Chức năng key phím 4×4:
      * + R1 : Ngõ ra hàng 1.
        + R2 : Ngõ ra hàng 2.
        + R3 : Ngõ ra hàng 3.
        + R4 : Ngõ ra hàng 4.
        + C1 : Ngõ vào hàng 1.
        + C2 : Ngõ vào hàng 2.
        + C3 : Ngõ vào hàng 3.
        + C4 : Ngõ vào hàng 4.

## Cảm Biến Thân Nhiệt Chuyển Động PIR HC-SR501

|  |  |
| --- | --- |
| Ảnh có chứa biểu đồ, bản phác thảo, hình vẽ, hàng  Mô tả được tạo tự động | Ảnh có chứa bánh xe, đồ chơi, phương tiện  Mô tả được tạo tự động |

### Hình 2.5: Cảm Biến Thân Nhiệt Chuyển Động PIR HC-SR501

* + - Nguyên tắc hoạt động:
      * Cảm biến HC-SR501 hoạt động dựa trên nguyên lý cảm biến hồng ngoại không hoạt động.
      * Nó sử dụng các cảm biến PIR để phát hiện sự chuyển động trong lĩnh vực quét của nó.
    - Đặc điểm kỹ thuật cơ bản:
      * Khoảng cách phát hiện: Thường có thể phát hiện chuyển động trong khoảng từ vài mét đến vài chục mét, tùy thuộc vào mô hình cụ thể.
      * Thời gian hoạt động: Có thể điều chỉnh thời gian hoạt động sau khi phát hiện chuyển động, thường từ một vài giây đến vài phút.
      * Góc quét: Cảm biến thường có một góc quét rộng, cho phép phát hiện chuyển động trong một diện tích lớn.
    - Chân kết nối:
      * HC-SR501 thường có ba chân kết nối chính: VCC (nguồn), GND (đất), và OUT (đầu ra).
      * Nó có thể được nối với các vi điều khiển như Arduino thông qua các chân kết nối này.
    - Chức năng đặc biệt:
      * Điều chỉnh độ nhạy: Một số mô hình có thể được điều chỉnh để tăng hoặc giảm độ nhạy của cảm biến.
      * Điều chỉnh thời gian hoạt động: Người dùng có thể điều chỉnh thời gian mà cảm biến sẽ giữ trạng thái báo động sau khi phát hiện chuyển động.

## Servo SG90

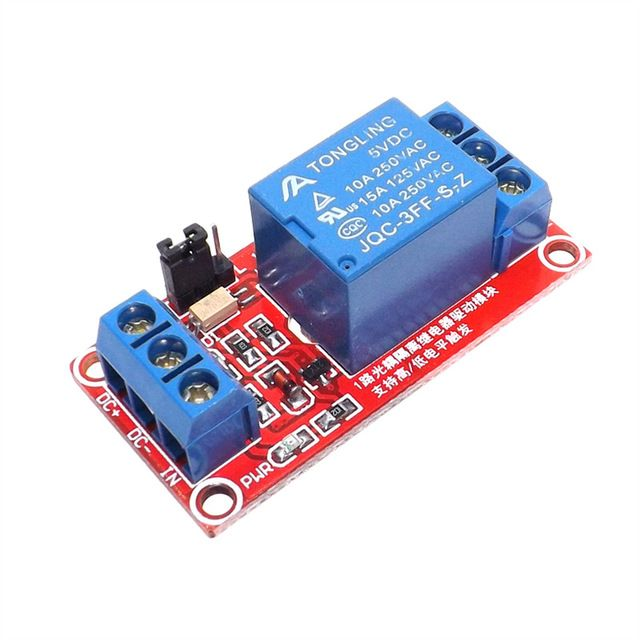
Ảnh có chứa nhựa, dây cáp, Xanh điện, màu xanh lam

Mô tả được tạo tự động

### Hình 2.6: Động cơ Servo SG90

* + - Servo là một dạng động cơ điện đặc biệt. Không giống như động cơ thông thường cứ cắm điện vào là quay liên tục, servo chỉ quay khi được điều khiển (bằng xung PPM) với góc quay nằm trong khoảng bất kì từ 0o - 180o. Mỗi loại servo có kích thước, khối lượng và cấu tạo khác nhau. Có loại thì nặng chỉ 9g (chủ yếu dùng trên máy bay mô mình), có loại thì sở hữu một momen lực bá đạo (vài chục Newton/m), hoặc có loại thì khỏe và nhông sắc chắc chắn,...
    - Động cơ servo được thiết kế những hệ thống hồi tiếp vòng kín. Tín hiệu ra của động cơ được nối với một mạch điều khiển. Khi động cơ quay, vận tốc và vị trí sẽ được hồi tiếp về mạch điều khiển này. Nếu có bầt kỳ lý do nào ngăn cản chuyển động quay của động cơ, cơ cấu hồi tiếp sẽ nhận thấy tín hiệu ra chưa đạt được vị trí mong muốn. Mạch điều khiển tiếp tục chỉnh sai lệch cho động cơ đạt được điểm chính xác. Các động cơ servo điều khiển bằng liên lạc vô tuyến được gọi là động cơ servo RC (radio-controlled). Trong thực tế, bản thân động cơ servo không phải được điều khiển bằng vô tuyến, nó chỉ nối với máy thu vô tuyến trên máy bay hay xe hơi. Động cơ servo nhận tín hiệu từ máy thu này.
    - Nguyên tắc hoạt động
      * Động cơ servo sử dụng một hệ thống phản hồi để duy trì vị trí cố định hoặc thực hiện chuyển động theo yêu cầu.
      * Bên trong động cơ servo thường có một bộ cảm biến (thường là potentiometer) để đo góc quay hoặc vị trí của trục động cơ.
    - Chân kết nối
      * Nguồn điện: Động cơ servo thường được cung cấp điện áp DC, và có thể yêu cầu điện áp cụ thể tùy thuộc vào mô hình.
      * Chân điều khiển: Có thể điều khiển thông qua các chân điều khiển như PWM (Pulse Width Modulation) từ các vi điều khiển như Arduino.

## Modul Relay



### Hình 2.7: Modul Relay 1 kênh(5-10V)

* + - Module Relay 1 kênh là một thiết bị điện tử được sử dụng để kiểm soát các thiết bị điện khác thông qua một tín hiệu điều khiển, thường là từ một vi điều khiển như Arduino hoặc Raspberry Pi. Dưới đây là một giới thiệu về một Module Relay 1 kênh với dải điện áp từ 5V đến 10V:
    - Nguyên tắc hoạt động:
      * Module Relay hoạt động dựa trên nguyên tắc sử dụng một cuộn dây cuộn để tạo ra một từ trường khi có điện áp được cấp.
      * Khi từ trường này được tạo ra, nó làm chuyển động một cơ cấu cơ khí hoặc điện từ để mở hoặc đóng các tiếp điểm relay.
    - Cấu trúc cơ bản:
      * Tiếp điểm Relay: Module này thường có một hoặc nhiều tiếp điểm relay, có thể là Normally Open (NO), Normally Closed (NC) hoặc có thể là cả hai.
      * Đèn LED báo trạng thái: Có thể có một đèn LED tích hợp để báo hiệu trạng thái hoạt động của relay.
    - Chân kết nối:
      * VCC và GND: Chân cấp nguồn cho relay.
      * IN (hoặc Signal): Chân này dùng để kết nối với nguồn điều khiển như Arduino, Raspberry Pi để kiểm soát relay.

## Điện trở.

### Hình 2.8: Điện trở dạng cắm

* + - Điện trở là linh kiện thụ động có tác dụng cản trở cả dòng và áp. Điện trở đựơc sử dụng rất nhiều trong các mạch điện tử.
    - Điện trở của dây dẫn có trị số điện trở lớn hay nhỏ tùy thuộc vào vật liệu làm dây, tỉ lệ thuận với chiều dài và tỉ lệ nghịch với tiết diện dây dẫn.

## 2.9 Led.



### Hình 2.9: Bóng led đủ màu

* + - Bóng đèn LED trong Arduino thường được sử dụng như một nguồn sáng dễ kết nối và kiểm soát. Bằng cách sử dụng chân digital output, người dùng có thể bật tắt bóng đèn LED và thậm chí thực hiện các hiệu ứng ánh sáng đơn giản trong các dự án điện tử.

## 2.10 Nút nhấn (Button)



### Hình 2.10: Button

* + - Nút nhấn (button) trong Arduino là một cảm biến đơn giản được sử dụng để nhận tín hiệu từ người dùng. Khi được nhấn, nút nhấn tạo ra một tín hiệu digital mà Arduino có thể sử dụng để kích thích các hành động hoặc điều khiển trong các dự án điện tử.

## 2.11 Ứng dụng quản lý trên điện thoại di động

# CHƯƠNG III: THIẾT KẾ VÀ THI CÔNG MẠCH

## Sơ đồ khối.

**KHỐI NGUỒN**

**KHỐI HIỂN THỊ**

**KHỐI HỆ NHÚNG**

**KHỐI THỜI GIAN THỰC**

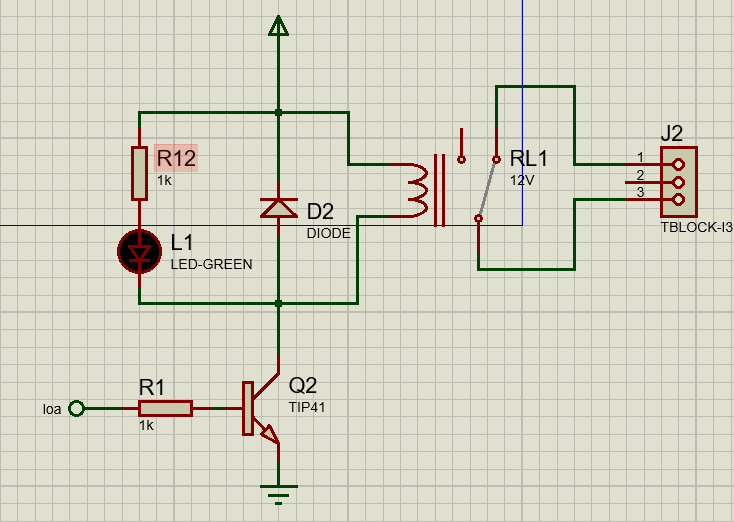
**KHỐI CÔNG SUẤT**

***Hình 3.1: Sơ đồ khối***

## Sơ đồ nguyên lý và chức năng của từng khối.

## Khối công suất

Do cuộn hút của chuông điện sử dụng nguồn điện xoay chiều 220VAC nên ta dụng Tranzitor điều khiển cuộn hút relay hoặc công tắc tơ, relay và công tắc tơ có tác dụng cách li về điện với mạch động lực và nó điều khiển đóng ngắt chuông điện.



### Hình 3.2: Khối công suất

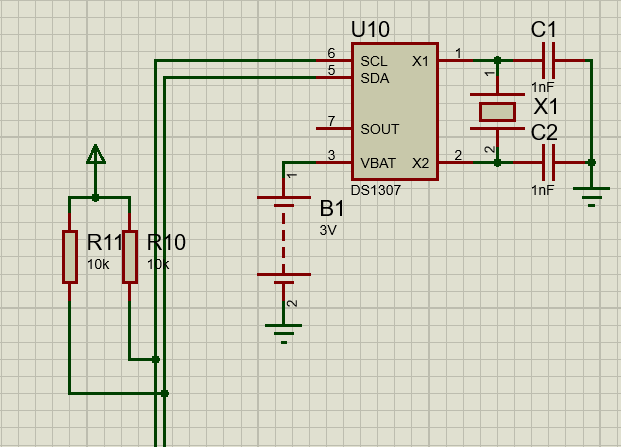
* Nguyên lý hoạt động:
  + Khi tín hiệu đưa vào là mức 0 (Tức =0V) thì Q2 không dẫn do không có dòng IBE >> Relay không làm việc.
  + Khi tín hiệu đưa vào là mức (Tức =5V) thì sẽ qua R hạn dòng làm cho Q2 dẫn thông lúc này ta có dòng Ice là dòng điện chạy qua cuộn dây >> Q2 >> MASS, Relay( 2V) đóng tiếp điểm thường mở (điều khiển thiết bị nào đó).
  + Diode D2 trong mạch có tác dụng chống lại dòng điện cảm ứng do cuộn đây sinh ra làm hỏng tranzitor. Relay dùng để điều khiển chuông điện 220V
  + Mục đích của R là tạo dòng vào cực B của trans tới ngưỡng bão hòa để trans hoạt động như chiếc khóa có điều kiện.

## Khối hệ nhúng(Atmega16)

### Hình 3.3: Khối hệ nhúng

* Các chân 1-8 thuộc PORTB của hệ nhúng, trong đó từ chân 1-3 được nối với khối nút nhấn để tao tín hiệu và điều chỉnh thông số trên LCD16X2 còn chân số 4 nối với khối công suất để điều khiển chuông 220V.
* Các chân 14-21 thuộc PORTD của hệ nhúng là ngõ ra của LCD 16x2.
* Hệ nhúng muốn hoạt động được cần có một nguồn tạo dao động. Trong các mạch hệ nhúng thường sử dụng thạch anh để tạo dao động.
* Để tăng độ ổn định tần số, người ta dùng thêm 2 tụ nhỏ C4, C5 (33pF x2), tụ bù nhiệt ổn tần.

## Khối thời gian thực

* + - * VCC,GND: nguồn 1 chiều được cung cấp tới các chân này. VCC là đầu vào 5V. Khi 5V được cung cấp thì thiết bị đó có thể truy cập hoàn chỉnh và dữ liệu có thể đọc và viết.
      * Khi pin 3V được kết nối tới thiết và vcc nhỏ hơn .25Vbat thì quá trình đọc và viết không được thực thi, tuy nhiên chức năng timekeeping không bị ảnh hưởng bởi điện áp vào thấp khi VCC nhỏ hơn Vbat thì RAM và time keeper sẽ được ngắt tới nguồn cung cấp( 3-5VDC).

### Hình 3.4: Khối thời gian thực

* Vbat: đầu vào pin cho bất kỳ một chuẩn pin 3V. Điện áp pin phải giữ trong khoảng 2.5-3V để đảm bảo cho thiết bị hoạt động tốt.
* SCL(serial clock input): SCL được xử dụng để đồng bộ sự chuyển dữ liệu trên đường dây nối tiếp.
* SDA(serial data input/output): là chân ra vào cho 2 đường dây nối tiếp. chân SDA được thiết kế theo kiêu cực máng hở, vì vậy phải có điện trở R10=10K và R11=10K kéo lên trong khi hoạt động.
* X1,X2: được nối với thạch anh với tần số 32,768 kHz. là một mạch tạo dao động ngoài, để hoạt động ổn định ta nói thêm 2 tụ C1,C2 với giá trị 33pF.

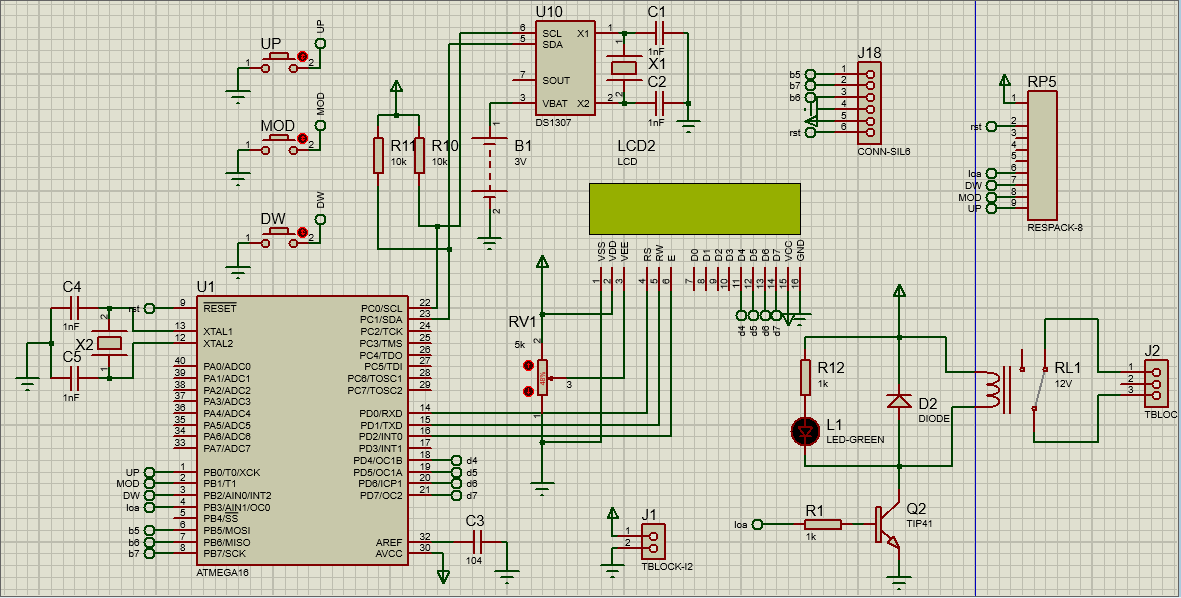
## Khối hiển thị LCD 16X2

### Hình 3.5: Khối hiển thị

* Các chân 1,2,3 là các chân VSS , VDD, VEE trong đó VSS chân nối đất , VEE chân chọn độ tương phản chân này dc chọn qua 1 biến trở 5K một đầu nối VEE

, một đầu nối mass chân VSS, một đầu nối nguồn chân VDD.

* Các chân D4, D5, D6, D7 được nối lần lượt với PORTD.4, PORTD.5, PORTD.6, PORTD.7
* Khối hiển thị dùng để hiển thị các thông tin: giờ, phút, giây và thứ, ngày, tháng, năm. Có 3 nút nhấn để điều chỉnh các thời gian hiển thị trên LCD.

***Hình 3.6: Sơ đồ nguyên lý chung***

## Thi công.

## Mạch in.

***Hình 3.7: Sơ đồ mạch in***

## Mạch hoàn chỉnh



***Hình 3.8: Hình ảnh mạch hoàn chỉnh***

## Code chƣơng trình hệ nhúng:

#include <mega16.h> #include <delay.h> #include <ds1307.h> #include <alcd.h>

#define up PINB.2 #define mod PINB.1 #define dw PINB.0 #define loa PORTB.3

unsigned char h,m,s,index; unsigned char ngay,thang,nam,thu;

void hienthi()

{

lcd\_gotoxy(0,0); lcd\_puts("TIME: "); lcd\_putchar(h/10+0x30); lcd\_putchar(h%10+0x30); lcd\_puts(":"); lcd\_putchar(m/10+0x30); lcd\_putchar(m%10+0x30); lcd\_puts(":"); lcd\_putchar(s/10+0x30); lcd\_putchar(s%10+0x30);

lcd\_gotoxy(0,1); if(thu==1)

{lcd\_putsf("SUN");}

if(thu==2)

{lcd\_putsf("MON");} if(thu==3)

{lcd\_putsf("TUE");} if(thu==4)

{lcd\_putsf("WED");} if(thu==5)

{lcd\_putsf("THU");} if(thu==6)

{lcd\_putsf("FRI");} if(thu==7)

{lcd\_putsf("SAT");}

lcd\_gotoxy(6,1); lcd\_putchar(ngay/10+0x30); lcd\_putchar(ngay%10+0x30); lcd\_puts("/"); lcd\_putchar(thang/10+0x30); lcd\_putchar(thang%10+0x30); lcd\_puts("/20"); lcd\_putchar(nam/10+0x30); lcd\_putchar(nam%10+0x30);

}

void chinh()

{

delay\_ms(1); if(mod==0)

{

index++; if(index>6)index=0;

while(mod==0);

}

}

void nutnhan()

{

delay\_ms(1); if(index==1)

{

if(up==0)

{

h++;

if(h>23)h=0; rtc\_set\_time(h,m,s); while(up==0);

}

if(dw==0)

{

h--; if(h==255)h=23;

rtc\_set\_time(h,m,s); while(dw==0);

}

}

if(index==2)

{

if(up==0)

{

m++;

if(m>59)m=0;

rtc\_set\_time(h,m,s); while(up==0);

}

if(dw==0)

{

m--; if(m==255)m=59;

rtc\_set\_time(h,m,s); while(dw==0);

}

}

if(index==3)

{

if(up==0)

{

thu++; if(thu>7)thu=1;

rtc\_set\_date(thu,ngay,thang,nam); while(up==0);

}

if(dw==0)

{

thu--; if(thu==0)thu=7;

rtc\_set\_date(thu,ngay,thang,nam); while(dw==0);

}

}

if(index==4)

{

if(up==0)

{

ngay++; if(ngay>31)ngay=1;

rtc\_set\_date(thu,ngay,thang,nam); while(up==0);

}

if(dw==0)

{

ngay--; if(ngay==0)ngay=31;

rtc\_set\_date(thu,ngay,thang,nam); while(dw==0);

}

}

if(index==5)

{

if(up==0)

{

thang++; if(thang>12)thu=1;

rtc\_set\_date(thu,ngay,thang,nam); while(up==0);

}

if(dw==0)

{

thang--; if(thang==0)thang=12;

rtc\_set\_date(thu,ngay,thang,nam); while(dw==0);

}

}

if(index==6)

{

if(up==0)

{

nam++; if(nam>99)nam=0;

rtc\_set\_date(thu,ngay,thang,nam); while(up==0);

}

if(dw==0)

{

nam--; if(nam==255)nam=99;

rtc\_set\_date(thu,ngay,thang,nam); while(dw==0);

}

}

}

void keu\_chuong\_3tieng()

{

loa=1; delay\_ms(200); loa=0; delay\_ms(200);

loa=1;

delay\_ms(200); loa=0; delay\_ms(200);

loa=1; delay\_ms(200); loa=0; delay\_ms(200);

}

void kiemtra\_bao\_gio()

{

if(thu>1)

{

if((h==7)&&(m==30))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==3) keu\_chuong\_3tieng(); else if(s==6) keu\_chuong\_3tieng();

}

else if((h==9)&&(m==45))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==4) keu\_chuong\_3tieng(); else if(s==8) keu\_chuong\_3tieng();

}

else if((h==10)&&(m==00))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==3) keu\_chuong\_3tieng();

else if(s==6) keu\_chuong\_3tieng();

}

else if((h==11)&&(m==30))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==4) keu\_chuong\_3tieng(); else if(s==8) keu\_chuong\_3tieng();

}

else if((h==12)&&(m==30))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==3) keu\_chuong\_3tieng(); else if(s==6) keu\_chuong\_3tieng();

}

else if((h==14)&&(m==00))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==4) keu\_chuong\_3tieng(); else if(s==8) keu\_chuong\_3tieng();

}

else if((h==14)&&(m==15))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==3) keu\_chuong\_3tieng(); else if(s==6) keu\_chuong\_3tieng();

}

else if((h==16)&&(m==30))

{

if(s==0) keu\_chuong\_3tieng(); else if(s==4) keu\_chuong\_3tieng();

else if(s==8) keu\_chuong\_3tieng();

}

}

}

void main(void)

{

DDRD=0xff; DDRB=0xf8; DDRC=0x04;

lcd\_init(16); i2c\_init();

// rtc\_get\_time(&h,&m,&s); rtc\_set\_date(6,07,06,19); rtc\_set\_time(02,53,00); dem=0;

index=0; while (1)

{

rtc\_get\_time(&h,&m,&s); rtc\_get\_date(&thu,&ngay,&thang,&nam); chinh();

nutnhan(); nhap\_nhay\_chinh(); kiemtra\_bao\_gio();

}

}

# CHƢƠNG IV: KẾT QUẢ THỰC HIỆN VÀ KẾT LUẬN

## Ƣu , nhƣợc điểm của sản phẩm

* + - Ưu điểm:
      * Dễ dàng sử dụng.
      * Giá thành rẻ, dễ thi công , dễ sửa chữa , độ an toàn cao.
      * Độ chính xác cao.
    - Nhược điểm:
      * Không điều chỉnh được thời gian hẹn giờ.
      * Bố trí linh kiện chưa được khoa học.

## Kết luận.

Trong quá trình nghiên cứu và xây dựng đề tài em đã vận dụng những kiến thức chuyên môn. nhằm để hoàn thành yêu cầu đặt ra. Qua đó em được dịp cũng cố lại kiến thức cơ bản chuyên ngành. Trên cơ sơ đó chúng em có thể đánh giá lại những gì mình có được sau khi học.

Trong thời gian thực hiện đề tài với sự chỉ bảo và giúp đỡ tận tình của giảng viên hướng dẫn đến nay mạch chuông báo tiết học đã hoàn thành.Tuy nhiên do thời gian có hạn và trình độ chuyên môn còn hạn chế nên đồ án còn tồn tại những thiếu sót. Em rất mong nhận được những ý kiến và góp ý của quý thầy cô trong khoa để sản phẩm của em được hoàn thiện hơn. Cuối cùng chúng em xin được cảm ơn quý thầy cô giáo đã tạo điều kiện tốt nhất giúp đỡ chúng em hoàn thành đề tài.

Em xin chân thành cảm ơn!

**Tiếng Việt**

# TÀI LIỆU THAM KHẢO

[1]. Lâm Quang Chuyên, 2012, *Giáo trình hệ nhúng*, Trường Cao đẳng Công thương Tp. HCM.

[2]. Nguyễn Đình Phú, Nguyễn Trường Duy, 2013, *Kỹ Thuật Số*, Xuất bản Đại Học Quốc Gia, Tp.HCM.

**Tiếng Anh**

[1]. Brander, J., 1985a, *Competition Management*, Journal of International Economics*,* Số 18*,* trang 83-100.

[2]. Brander, J., 1985b, *Benefits of Competition*, Journal of International Economics*,* Số 18*,* trang 68-108.

[3]. Jaffe, J. and Westerfield R., 1985, *The impact of inflation,* Journal of Finance, Số 40, trang 25-34.

[4]. UNDP, 2011, *Human Development Report.*

[*https://khuenguyencreator.com/tong-quan-ve-so-do-chan-esp32-va-ngoai-vi/*](https://khuenguyencreator.com/tong-quan-ve-so-do-chan-esp32-va-ngoai-vi/)

*http://arduino.vn/bai-viet/42-arduino-uno-r3-la-gi*