

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN-ĐIỆN TỬ



HCMUTE

ĐỒ ÁN MÔN HỌC 1

Đề tài: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG
ĐO NỒNG ĐỘ CÒN HIỂN THỊ TRÊN LED 7 ĐOẠN

SVTH: PHAN MINH QUÂN
MSSV: 21161353
Khóa: 2021
Ngành: CNKT Điện tử - Viễn thông
GVHD: TS. NGUYỄN NGÔ LÂM

Tp. Hồ Chí Minh, tháng 6 năm 2024



ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
**KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO**
www.fhq.hcmute.edu.vn

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---***---

Tp. Hồ Chí Minh, ngày 15 tháng 06 năm 2024

NHIỆM VỤ ĐỒ ÁN MÔN HỌC

Họ và tên sinh viên: Phan Minh Quân

MSSV: 21161353

Ngành: Công Nghệ Kỹ Thuật Điện tử - Viễn thông

Lớp: 21161CLVT2A

Giảng viên hướng dẫn: TS. Nguyễn Ngô Lâm

Ngày nhận đề tài:

Ngày nộp đề tài:

1. Tên đề tài: Thiết kế và thi công hệ thống đo nồng độ cồn hiển thị trên led 7 đoạn

2. Các số liệu, tài liệu ban đầu:

Kiến thức cơ bản về các môn Mạch điện, Điện tử cơ bản, Vi xử lý, Arduino.

3. Nội dung thực hiện đề tài:

- Thiết kế hệ thống
- Mô phỏng mạch trên Proteus
- Lập trình cho hệ thống
- Chỉnh sửa và kiểm tra mạch
- Viết báo cáo

4. Sản phẩm:

Thiết kế và thi công hệ thống đo nồng độ cồn hiển thị trên led 7 đoạn

GIẢNG VIÊN HƯỚNG DẪN

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên Sinh viên: Phan Minh Quân

MSSV: 21161353

Ngành: Công Nghệ Kỹ Thuật Điện tử - Viễn thông

Tên đề tài: Thiết kế và thi công hệ thống đo nồng độ cồn hiển thị trên led 7 đoạn

Họ và tên Giáo viên hướng dẫn: TS. Nguyễn Ngô Lâm

NHẬN XÉT

1. Về nội dung đề tài & khối lượng thực hiện:

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

2. Ưu điểm:

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

3. Khuyết điểm:

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

4. Đề nghị cho bảo vệ hay không?

.....

5. Đánh giá loại:

.....

6. Điểm: (Bằng chữ)

.....

Tp. Hồ Chí Minh, ngày 15 tháng 06 năm 2024

Giáo viên hướng dẫn



ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM
KHOA ĐÀO TẠO
CHẤT LƯỢNG CAO
www.fhq.hcmute.edu.vn

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

---***---

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên Sinh viên: Phan Minh Quân

MSSV: 21161353

Ngành: Công Nghệ Kỹ Thuật Điện tử - Viễn thông

Tên đề tài: Thiết kế và thi công hệ thống đo nồng độ cồn hiển thị trên led 7 đoạn

Họ và tên Giáo viên phản biện:

NHẬN XÉT

1. Về nội dung đề tài & khối lượng thực hiện:

.....

.....

.....

.....

2. Ưu điểm:

.....

.....

.....

.....

3. Khuyết điểm:

.....

.....

4. Đề nghị cho bảo vệ hay không?

.....

5. Đánh giá loại:

.....

7. Điểm: (Bằng chữ)

.....

Tp. Hồ Chí Minh, ngày 15 tháng 6 năm 2024

Giáo viên phản biện

LỜI CẢM ƠN

Để hoàn thành báo cáo đồ án môn học 1 chuyên ngành Công nghệ Kỹ thuật Điện tử - Viễn thông trước hết em xin gửi đến quý Thầy/Cô trong khoa Đào tạo Chất lượng cao, trường Đại học Sư Phạm Kỹ Thuật Thành Phố Hồ Chí Minh lời cảm ơn chân thành. Đặc biệt, thầy **Nguyễn Ngô Lâm** đã tận tình hướng dẫn, giúp đỡ và tạo điều kiện thuận lợi cho em trong suốt quá trình thực hiện đồ án. Em xin gửi đến thầy lời cảm ơn chân thành và sâu sắc nhất.

Đồng thời, em cũng xin cảm ơn đến các bạn bè đã hỗ trợ, đóng góp ý kiến cũng như chia sẻ kinh nghiệm để em hoàn thành tốt đề tài.

Mặc dù đã cố gắng hết sức, nhưng do lượng kiến thức còn eo hẹp nên không tránh khỏi những thiếu sót. Do vậy nên em rất mong nhận được sự góp ý quý báu của Thầy/Cô để có thể hoàn thiện và tốt hơn nữa cũng như tích lũy kinh nghiệm để hoàn thành tốt báo cáo đồ án môn học 2 và báo cáo đồ án tốt nghiệp sau này.

Sau cùng, em kính chúc quý thầy cô thật dồi dào sức khỏe, luôn tràn đầy nhiệt huyết cùng với thành công trong sự nghiệp cao quý.

Em xin chân thành cảm ơn!

MỤC LỤC

TRANG

NHIỆM VỤ ĐỒ ÁN MÔN HỌC	
PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN	
PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN	
LỜI CẢM ƠN.....	i
MỤC LỤC.....	ii
DANH MỤC HÌNH ẢNH	iv
DANH MỤC CÁC BẢNG BIỂU	v
Chương 1: TỔNG QUAN	1
1.1 Giới thiệu đề tài.....	1
1.3. Đối tượng nghiên cứu.....	2
1.4. Phạm vi nghiên cứu.....	2
1.5. Bố cục đồ án.....	2
Chương 2: CƠ SỞ LÝ THUYẾT.....	4
2.1 Cảm biến nồng độ cồn.....	4
Chương 3: TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG.....	24
3.1 Yêu cầu và sơ đồ khối của hệ thống	24
3.1.1 Yêu cầu của hệ thống	24
3.1.2 Sơ đồ khối của hệ thống.....	24
3.2 Thiết kế phần cứng hệ thống.....	25
3.2.1 Khối nhận dữ liệu.....	25

3.2.2 Khối hiển thị.....	26
3.2.3 Khối nút nhấn.....	27
3.2.4 Khối báo động.....	27
3.2.5 Khối xử lý trung tâm.....	28
3.2.6 Khối nguồn.....	29
3.3 Thiết kế phần mềm hệ thống.....	29
3.4 Lưu đồ giải thuật của hệ thống.....	31
Chương 4: KẾT QUẢ_ NHẬN XÉT_ĐÁNH GIÁ	32
4.1 Kết quả sản phẩm.....	32
4.2 Kết quả toàn hệ thống hoạt động	32
Chương 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	34
5.1 Kết luận	34
5.2 Hạn chế và hướng phát triển	34
5.2.1 Hạn chế.....	34
5.2.2 Hướng phát triển	34
TÀI LIỆU THAM KHẢO.....	36
PHỤ LỤC.....	37

DANH MỤC HÌNH ẢNH

Hình 2.1.1: Cảm biến MQ-3

Hình 2.1-3: Bảng thể hiện giá trị của cảm biến thu được qua từng loại khí

Hình 2.2-1: Hình ảnh của module MQ3

Hình 2.3-1: hình ảnh thực tế của Led 7 đoạn

Hình 2.3-2: Sơ đồ chân của led 7 đoạn

Hình 2.3-3: sơ đồ Led 7 đoạn mắc anode chung

Hình 2.4-1: Hình ảnh của module TM1637

Hình 2.5-1: Bo mạch Arduino Uno R3

Hình 2.5-2: Các chân GPIO của Arduino Uno R3

Hình 2.5-3: Sơ đồ nguyên lý của Arduino Uno R3

Hình 2.6: Hình ảnh LED đơn

Hình 2.7: Hình ảnh Buzzer

Hình 2.8 : Hình ảnh thực tế của nút nhấn

Hình 3.1.2-1 Sơ đồ khối của hệ thống

Hình 3.2.1: Sơ đồ nối chân của module nồng độ cồn MQ-3

Hình 3.2.2: sơ đồ nguyên lý nối chân của Module_TM1637

Hình 3.2.3: Sơ đồ nguyên lý nối chân của khối nút nhấn

Hình 3.2.4: sơ đồ nguyên lý của khối báo động

Hình 3.3: Giao diện phần mềm Arduino IDE

Hình 3.4: lưu đồ giải thuật của hệ thống

Hình 4.1 Test module sản phẩm

Hình 4.2 Mô hình sản phẩm

DANH MỤC CÁC BẢNG BIỂU

Bảng 2.2-1: Bảng liệt kê và chức năng các chân của module cảm biến MQ-3

Bảng 2.2-2: Bảng đặc điểm độ nhạy của cảm biến (tính theo PPM)

Bảng 2.2-3: Bảng giá trị PPM quy đổi ra mức độ say của người uống rượu bia

Bảng 2.3-1: Bảng trạng thái của Led 7 đoạn mắc Anode chung

Bảng 2.3-2: Bảng trạng thái của Led 7 đoạn mắc cathode chung

Bảng 2.4: Bảng các thông số chân module TM1637

Bảng 2.5-1: Thông số của vi điều khiển Arduino Uno R3

Bảng 2.5-2: thông tin các chân analog của Arduino Uno R3

Bảng 2.5-3: thông tin các chân analog và các port nguồn của Arduino Uno R3

Bảng 3.2.6: Bảng tính toán nguồn hoạt động của hệ thống

Chương 1: TỔNG QUAN

1.1 Giới thiệu đề tài

Trong bối cảnh hiện nay, việc kiểm soát nồng độ cồn trong cơ thể con người đang trở thành một vấn đề cấp bách và cần thiết, đặc biệt trong các lĩnh vực an toàn giao thông, y tế và công nghiệp. Sự phát triển của công nghệ đã mở ra nhiều cơ hội mới trong việc thiết kế và ứng dụng các thiết bị đo lường chính xác, hiệu quả và tiện dụng. Điều đó cho thấy rằng việc "Thiết kế và thi công hệ thống đo nồng độ cồn" là rất quan trọng và cần thiết. Nó không những đáp ứng được nhu cầu thực tiễn kể trên mà còn góp phần nâng cao chất lượng đời sống cho người dân.

Với đề án này, em hi vọng sẽ xây dựng nên một hệ thống đo nồng độ cồn có độ chính xác cao, dễ sử dụng và có khả năng ứng dụng rộng rãi. Hệ thống này không chỉ giúp phát hiện nồng độ cồn trong cơ thể người dùng mà còn hỗ trợ trong việc đưa ra các biện pháp phòng ngừa và quản lý hiệu quả trong nhiều lĩnh vực chuyên môn khác.

Đề án sẽ được trình bày chi tiết các bước từ nghiên cứu, thiết kế, lắp ráp, đến thử nghiệm và hiệu chỉnh hệ thống. Bên cạnh đó, em cũng sẽ phân tích và đánh giá hiệu quả của hệ thống thông qua các thử nghiệm thực tế, từ đó đề xuất các cải tiến và ứng dụng tiềm năng trong tương lai.

Hy vọng rằng, với sự cố gắng và nỗ lực của em, đề án này sẽ đóng góp một phần nhỏ vào việc nâng cao chất lượng đời sống và là nền tảng cho những công trình nghiên cứu sau này.

1.2. Mục tiêu nghiên cứu

Nghiên cứu cơ sở lý thuyết về vi điều khiển Arduino, module hiển thị led 7 đoạn, cảm biến nồng độ cồn,

Nghiên cứu, thực hành các thao tác kỹ thuật điện tử cơ bản (lắp ráp, test mạch, mô phỏng, thiết kế...).

Nghiên cứu cách lập trình cho vi điều khiển, giao tiếp với module Led 7 đoạn, lấy dữ

liệu từ cảm biến.

1.3. Đối tượng nghiên cứu

Ở trong đề án 1 này em sẽ nghiên cứu về các đối tượng sau:

- Cảm biến đo nồng độ cồn.
- Vi điều khiển Arduino Uno R3.
- Module Led 7 đoạn TM1637.

Kết nối các module và cảm biến với board Arduino để hiển thị giá trị nồng độ cồn trên Led 7 đoạn.

1.4. Phạm vi nghiên cứu

- Nghiên cứu cơ sở lý thuyết về nồng độ cồn, giá trị nồng độ cồn cho phép . Mục đích của đề tài tạo ra sản phẩm có thể phát hiện nồng độ cồn chính xác, tối ưu, mang tính di động.
- Tìm hiểu các lý thuyết có liên quan như ngôn ngữ lập trình Arduino.

1.5. Bố cục đề án

Bố cục đề án 1 này gồm 5 chương:

Chương 1: Tổng quan: Nêu tính cấp thiết của đề tài, xu hướng và tình hình khoa học và công nghệ hiện nay. Sự phát triển công nghiệp và đời sống hằng ngày và từ đó đưa ra lý do chọn đề tài và xác định mục tiêu cho đề tài.

Chương 2: Cơ sở lý thuyết: Trình bày tổng quan về các thành phần và chức năng của từng loại phần cứng có trong hệ thống, dẫn dắt chi tiết cụ thể để xây dựng hoàn chỉnh về mô hình.

Chương 3: Thiết kế và xây dựng hệ thống: Từ yêu cầu đề tài, trình bày về sơ đồ hệ thống. Qua đó nêu ra các phương pháp xử lý dữ liệu rồi từ đó thiết kế mô hình.

Chương 4: Kết quả thực hiện: Trình bày kết quả hiển thị giá trị nồng độ cồn qua Led 7 đoạn bằng cách thiết kế ra mô hình. Đưa ra các hiển thị như mong muốn đã lập trình.

Chương 5: Kết luận và hướng phát triển: Dựa vào kết quả có được từ chương 4, đưa ra kết luận tổng quan về những gì đạt được và chưa đạt được của đề tài. Từ đó đưa ra hướng phát triển để cải thiện hệ thống.

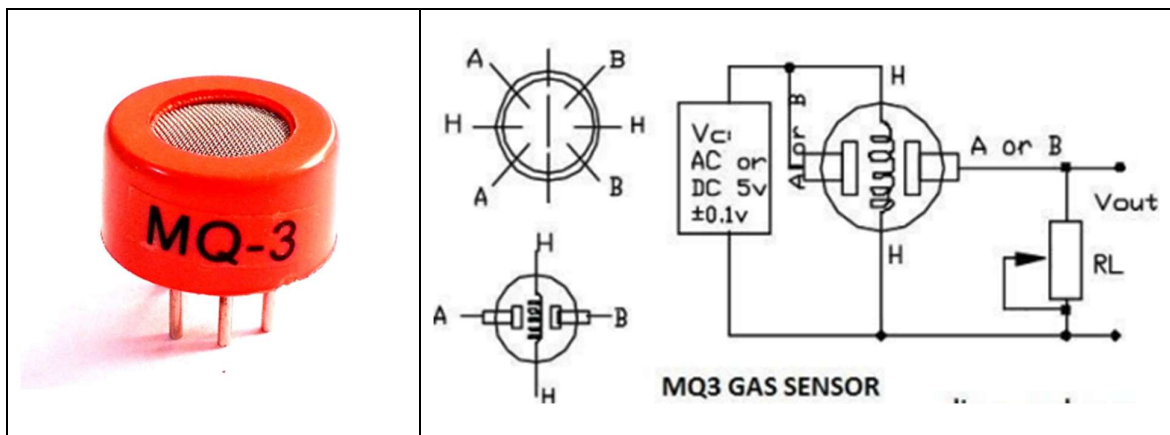
Chương 2: CƠ SỞ LÝ THUYẾT

2.1 Cảm biến nồng độ cồn

- **Khái niệm về nồng độ cồn**

Nồng độ cồn là một chỉ số đo hàm lượng cồn có trong một dung dịch hoặc thức uống có cồn và được tính theo phần trăm thể tích. Độ cồn được tính theo số nguyên chất mililit ethanol có trong 100mm dung dịch ở nhiệt độ 20 độ C. Ở một vài nước trên thế giới, độ cồn này còn được tính theo thang Gay-Lussac (đặt theo tên nhà hóa học Pháp Joseph Louis Gay - Lussac).

2.1.1 Cảm biến nồng độ cồn MQ3



Hình 2.1.1 Cảm biến MQ3

Cảm biến MQ-3 này được sử dụng nhằm mục đích đo nồng độ cồn có trong môi trường. Cảm biến này làm từ vật liệu SnO₂, loại vật liệu có tính dẫn điện kém trong môi trường không khí nhưng lại rất nhạy cảm với cồn. Nếu ở trong môi trường có nồng độ cồn càng cao thì điện trở của cảm biến này sẽ càng giảm.

- **Cấu tạo bên trong:**

Cảm biến MQ-3 gồm có 6 chân, vỏ và thân. Mặc dù nó có 6 chân, nhưng ta chỉ có thể sử dụng 4 chân. Hai trong số chúng là cho hệ thống đốt nóng, mà tôi gọi là H và 2 cái còn lại là để kết nối nguồn VCC và GND gọi là A và B.

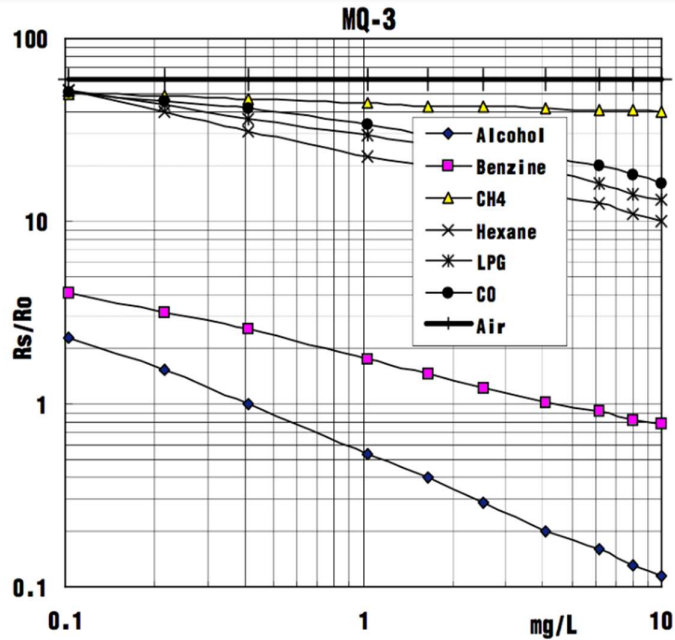
Nếu ta nhìn vào bên trong cảm biến thì sẽ tìm thấy ống nhỏ. Ống này là một hệ thống đốt nóng được làm từ nhôm oxit và thiếc điôxit và bên trong nó có các cuộn dây đốt nóng, dùng để tạo ra nhiệt. Ta cũng có thể nhìn thấy 6 chân, trong đó 2 chân là Chân H được kết nối với cuộn dây gia nhiệt và các chân khác được kết nối với ống.

- **Nguyên lý hoạt động cảm biến khí ga MQ3**

Cảm biến MQ3 hoạt động dựa trên nguyên lý điện trở. Cảm biến sử dụng một lớp màng mạ mỏng từ SnO₂ (Oxit Thiếc) làm vật liệu chính. Khi còn tiếp xúc với màng SnO₂, phản ứng hóa học xảy ra, làm thay đổi điện trở của màng.

Cảm biến MQ-3 có một bộ điều chỉnh tỷ lệ dòng điện để tạo điều kiện làm nóng cho màng SnO₂. Khi cấp điện cho cảm biến, dòng điện được chạy qua màng SnO₂, làm nóng màng lên nhiệt độ làm việc. Khi khí còn tiếp xúc với màng SnO₂, các phản ứng hóa học xảy ra và làm thay đổi điện trở của màng. Thay đổi điện trở này được đo và chuyển đổi thành tín hiệu analog hoặc digital để xác định mức độ còn trong không khí.

Điện trở của màng SnO₂ thay đổi tương ứng với nồng độ còn có mặt trong không khí. Cảm biến MQ-3 sẽ tạo ra một tín hiệu đầu ra có biên độ thay đổi phụ thuộc vào nồng độ còn. Từ đó, thông qua quá trình hiệu chỉnh và xử lý tín hiệu, có thể xác định được mức độ còn có trong môi trường xung quanh.

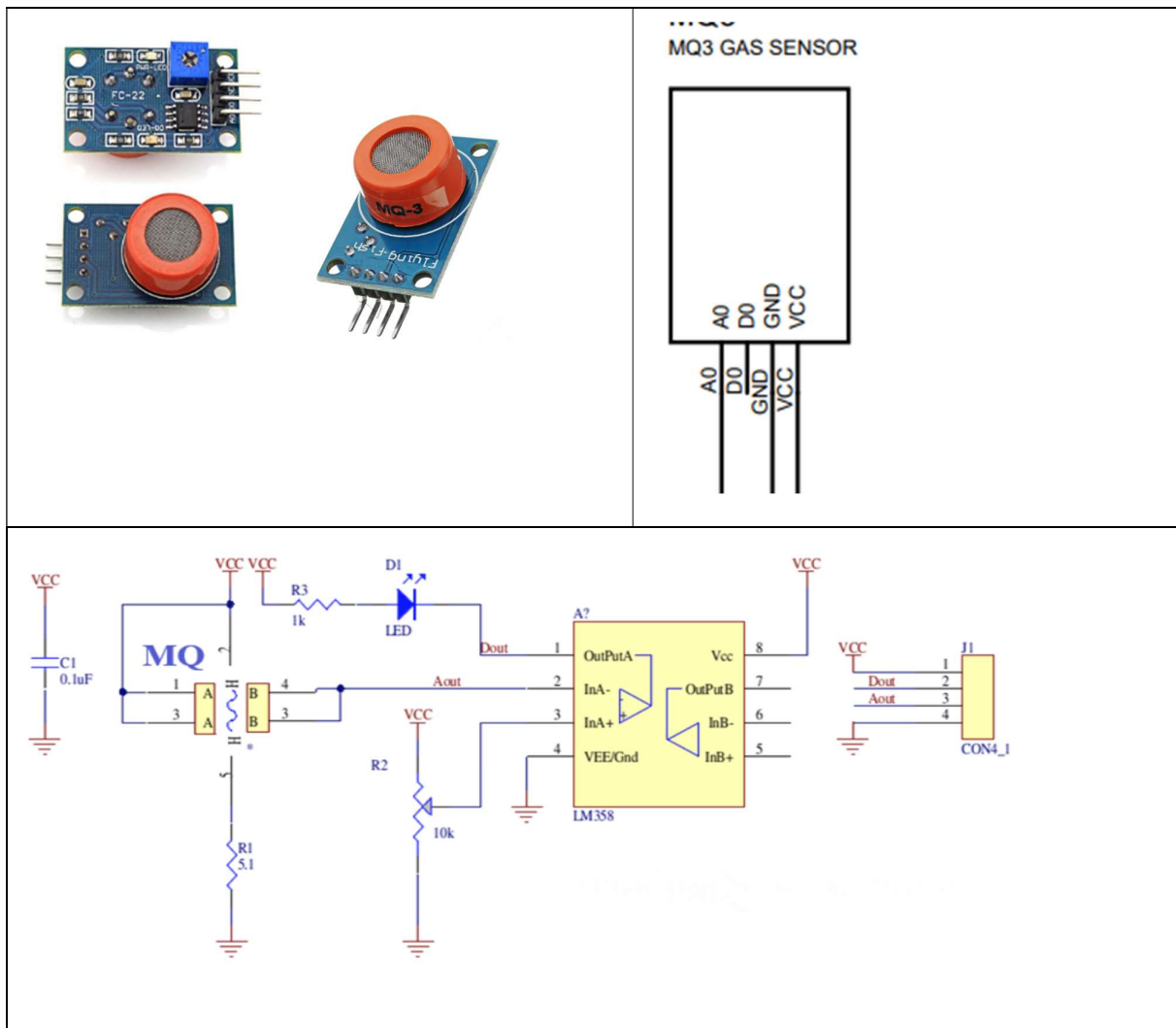


Hình 2.1-3: Bảng thể hiện giá trị của cảm biến thu được qua từng loại khí

2.2 Module cảm biến nồng độ cồn MQ3

Đây là loại module được tích hợp sẵn cảm biến nồng độ cồn MQ3. Module này dùng sử dụng để phát hiện nồng độ cồn trong môi trường, hơi thở. Cảm biến MQ3 được tích hợp có độ nhạy cao khả năng phản hồi nhanh, độ nhạy có thể điều chỉnh được bằng biến trở. Cảm biến cung cấp một đầu ra tương tự dựa trên nồng độ cồn.

Cảm biến MQ-3 có thể phát hiện khí gas, alcohol, ethanol.



Hình 2.2-1: Hình ảnh của module MQ3

Thông số kỹ thuật:

- Kích thước: 32 x 22 x 27 mm
- Chip chính: LM393, MQ-3 cảm biến khí.
- Điện áp cung cấp: DC 5V.
- Có 2 dạng tín hiệu đầu ra dạng Analog và TTL.
- Tín hiệu đầu ra TTL có giá trị thấp. (đầu ra tín hiệu mức thấp có thể được kết nối trực tiếp với vi điều khiển).

- Đầu ra tương tự 0 ~ 5V, nồng độ cồn càng cao điện áp càng cao.
- Độ nhạy cao và chọn lọc tốt với hơi ethanol.
- Bền và ổn định đáng tin cậy.
- Nhạy và phục hồi nhanh.

Chân	Tên	Đặc điểm
1	VCC	Nguồn cung cấp 3V hoặc 5V
2	GND	Ground
3	D0	Chân nhận tín hiệu chuyển sang dạng Digital
4	A0	Chân nhận tín hiệu chuyển sang Analog

Bảng 2.2-1: Bảng liệt kê và chức năng các chân của module cảm biến MQ-3

VOLATGE (V)	PPM (PART PER MILLION)	Tỷ lệ (%)
0	0	0
0.5	100	10
1	200	20
1.5	300	30
2	400	40
2.5	500	50
3	600	60
3.5	700	70
4	800	80
4.5	900	90
5	1000	100

Bảng 2.2-2: Bảng đặc điểm độ nhạy của cảm biến (tính theo PPM)

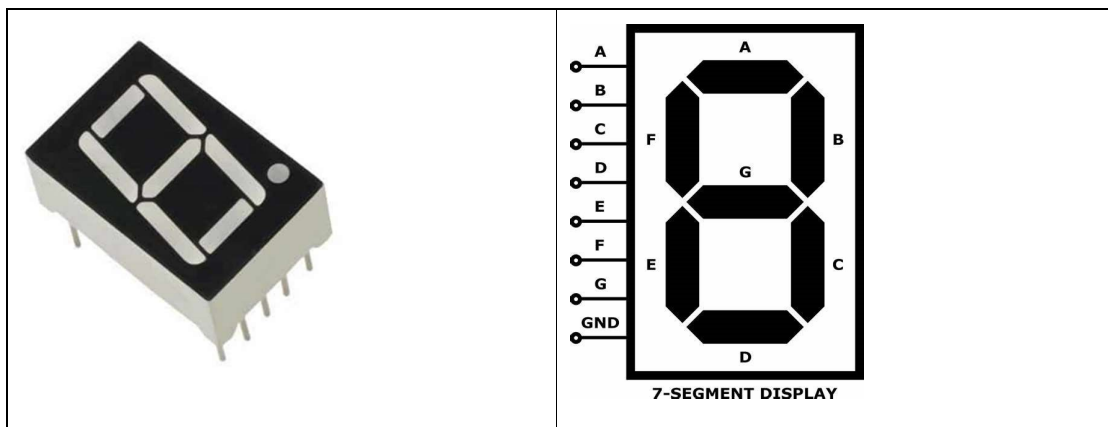
Giá trị PPM	Trạng thái
200 – 300	Hơi say
300 – 400	Say
400 - 500	Rất say

Bảng 2.2-3: Bảng giá trị PPM quy đổi ra mức độ say của người uống rượu bia

2.3 Led 7 đoạn

LED 7 đoạn là một module hiển thị kỹ thuật số được thiết kế đặc biệt để hiển thị thông tin số. Nó bao gồm một số lượng đèn LED được sắp xếp theo hình dạng của các con số, tạo ra một màn hình dễ dàng nhìn thấy. Mỗi con số trong màn hình được tạo ra bằng cách bật hoặc tắt các đèn LED tương ứng.

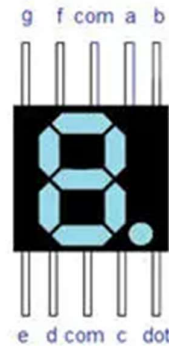
LED 7 đoạn thường được sử dụng trong các thiết bị hiển thị số như đồng hồ, bảng điện tử, thiết bị đo lường như máy đo nhiệt độ, công suất, áp suất và các ứng dụng điều khiển thời gian, đếm số, hiển thị kết quả, và thông báo lỗi. Với ưu điểm là cấu tạo đơn giản, giá thành rẻ và khả năng hiển thị các ký tự đơn giản, LED 7 đoạn là một phương pháp phổ biến và đáng tin cậy để hiển thị thông tin số trên các thiết bị điện tử.



Hình 2.3-1: hình ảnh thực tế của Led 7 đoạn

- **Cấu tạo của Led 7 đoạn**

Bên trong Led 7 đoạn gồm có 7 led đơn có dạng thanh và được xếp theo hình dạng số 8 có thể hiển thị được các số từ 0 – 9. Thêm vào đó sẽ có thêm một led đơn hình tròn nhỏ thể hiện dấu chấm tròn ở góc dưới.

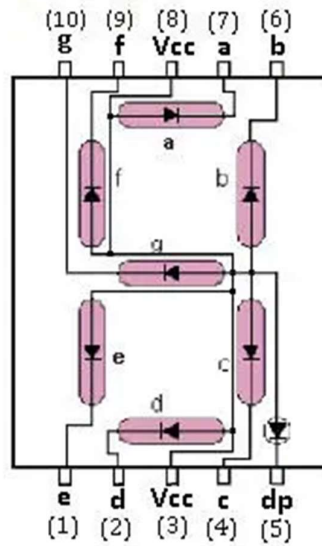


Hình 2.3-2: Sơ đồ chân của led 7 đoạn

LED 7 đoạn đơn gồm 10 chân đó là các chân: a, b, c, d, e, f, g, 2 chân chung và chân dấu chấm tròn. Như vậy nếu như muốn hiển thị ký tự nào thì ta chỉ cần xuất tín hiệu điều khiển vào chân đó là led sẽ sáng như mong muốn.

Led 7 đoạn sẽ gồm có 2 cách mắc phổ biến đó là mắc Anode chung (cực dương chung) và cathode chung (cực âm chung).

Với mắc anode chung thì chân 3 và chân 8 sẽ mắc lên Vcc với nhau, do đó nên ta muốn led nào đó sáng thì cực cathode của led đó phải được tích mức thấp .



Hình 2.3-3: sơ đồ Led 7 đoạn mắc anode chung

Còn ở cách mắc cathode chung thì Chân 3 và 8 là 2 chân chung được đất GND vậy nên nếu muốn led nào đó sáng thì cực anode của led đó phải được tích mức cao.

- **Nguyên lí hoạt động của Led 7 đoạn:**

Để tạo ra các chữ số 0-9, các đoạn LED cần được bật sáng theo các mẫu khác nhau. Ví dụ để hiển thị số “1” thì chỉ cần bật đoạn LED thứ 2 từ trên xuống, cho số “4” thì bật các đoạn 3, 5, 6. Việc chọn bật tắt các đoạn LED được điều khiển bởi một mạch điện tử bên trong.

Như vậy, bản chất hoạt động của LED 7 đoạn là sử dụng các mạch điện để điều khiển từng đoạn LED bật sáng theo một thứ tự nhất định, tạo thành các chữ số mong muốn. Kỹ thuật này giúp tạo ra các thiết bị hiển thị số với cấu tạo đơn giản và chi phí thấp.

Số	Số nhị phân								HEX
	7	6	5	4	3	2	1	0	
	dp	g	f	e	d	c	b	a	
0	1	1	0	0	0	0	0	0	C0
1	1	1	1	1	1	0	0	1	F9
2	1	0	1	0	0	1	0	0	A4
3	1	0	1	1	0	0	0	0	B0
4	1	0	0	1	1	0	0	1	99
5	1	0	0	1	0	0	1	0	92
6	1	0	0	0	0	0	1	0	82
7	1	1	1	1	1	0	0	0	8F
8	1	0	0	0	0	0	0	0	80
9	1	0	0	1	0	0	0	0	90
A	1	0	0	0	1	0	0	0	88
B	1	0	0	0	0	0	1	1	83
C	1	1	0	0	0	1	1	0	C6
D	1	0	1	0	0	0	0	1	A1
E	1	0	0	0	0	1	1	0	86
F	1	0	0	0	1	1	1	0	8E

Bảng 2.3-1: Bảng trạng thái của Led 7 đoạn mắc Anode chung

Số	Số nhị phân								HEX
	7	6	5	4	3	2	1	0	
	dp	g	f	e	d	c	b	a	
0	0	0	1	1	1	1	1	1	0x3F
1	0	0	0	0	0	1	1	0	0x06
2	0	1	0	1	1	0	1	1	0x5B
3	0	1	0	0	0	0	0	0	0x40
4	0	1	1	0	0	1	1	0	0x66
5	0	1	1	0	1	1	0	1	0x6D
6	0	1	1	1	1	1	0	1	0x7D
7	0	0	0	0	0	1	1	1	0x07
8	0	1	1	1	1	1	1	1	0x7F
9	0	1	1	0	1	1	1	1	0x6F
A	0	1	1	1	0	1	1	1	0x77
B	0	1	1	1	1	1	0	0	0x7C
C	0	0	1	1	1	0	0	1	0x39
D	0	1	0	1	1	1	1	0	0x5E
E	0	1	1	1	1	0	0	1	0x79
F	0	1	1	1	0	0	0	1	0x71

Bảng 2.3-2: Bảng trạng thái của Led 7 đoạn mắc cathode chung

Để điều khiển Led 7 đoạn ta có thể dùng các phương pháp sau:

- **Kết nối các chân điều khiển của LED trực tiếp với PORT của vi điều khiển**

Phương pháp này sẽ kết nối trực tiếp các chân điều khiển của LED 7 đoạn với các chân PORT của vi điều khiển. Phương pháp này đòi hỏi vi điều khiển có đủ số lượng chân điều khiển để kết nối với tất cả các đoạn của LED 7 đoạn.

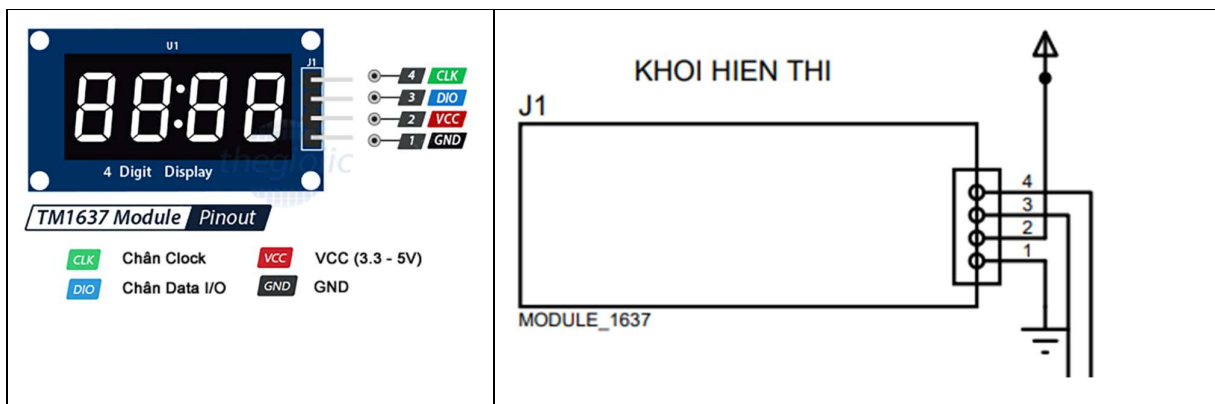
- **Phương pháp quét Led**

Trong một khoảng thời gian ngắn, chỉ có một đoạn LED được kích hoạt và hiển thị. Sau đó, đoạn tiếp theo được kích hoạt và thế trạng thái hiển thị trước đó được tắt đi. Quá trình này sẽ lặp lại với tần suất cao hơn tần số mà mắt người có thể nhận dạng được vật thể chuyển động khiến cho người sử dụng có cảm giác các đoạn đang được hiển thị cùng nhau.

- **Sử dụng quét LED nhưng dùng IC chốt dữ liệu**

Phương pháp này sử dụng một IC chốt dữ liệu để tiết kiệm các chân điều khiển của LED 7 đoạn. Khi sử dụng IC chốt dữ liệu, người dùng chỉ cần một số ngõ vào của vi điều khiển để gửi dữ liệu tới IC chốt, từ đó điều khiển hiển thị của LED 7 đoạn. Điều này giúp giảm đáng kể số lượng chân điều khiển cần thiết và tối ưu hóa việc lập trình và điều khiển LED 7 đoạn.

2.4 Module led 7 đoạn Tm1637



Hình 2.4-1: Hình ảnh của module TM1637

Module TM1637 là một module hiển thị LED sử dụng chip điều khiển TM1637, thường được sử dụng để hiển thị số và ký tự trên các thiết bị điện tử. Module này có chức năng chính là dùng để hiển thị các số và ký tự trên màn hình LED 7 đoạn. Việc tích hợp sẵn module này giúp cho kết nối led 7 đoạn trở nên dễ dàng và gọn gàng hơn. Các đặc điểm của module như sau:

- Điện áp hoạt động: 3.3V - 5V
- Dòng điện hoạt động: Khoảng 80mA (tùy thuộc vào số lượng đoạn LED được bật)
- Số chân kết nối: 4 chân (VCC, GND, DIO, CLK), trong đó:
 - VCC: Cấp nguồn cho module (3.3V - 5V)
 - GND: Chân nối đất
 - DIO (Data In/Out): Chân dữ liệu để truyền và nhận thông tin
 - CLK (Clock): Chân xung nhịp để đồng bộ hóa dữ liệu
- Nhiệt độ hoạt động: -40°C đến +85°C
- Tần số giao tiếp: Có thể đạt đến 500kHz
- Loại giao tiếp: Giao tiếp 2 dây (Clock và Data)
- Độ sáng: Có thể điều chỉnh độ sáng của LED qua phần mềm

Chân	Tên	Đặc điểm
1	Vcc	Nguồn cung cấp 3V hoặc 5V
2	GND	Ground
3	DIO	Đầu vào/ra dữ liệu
4	CLK	Tín hiệu xung CLOCK

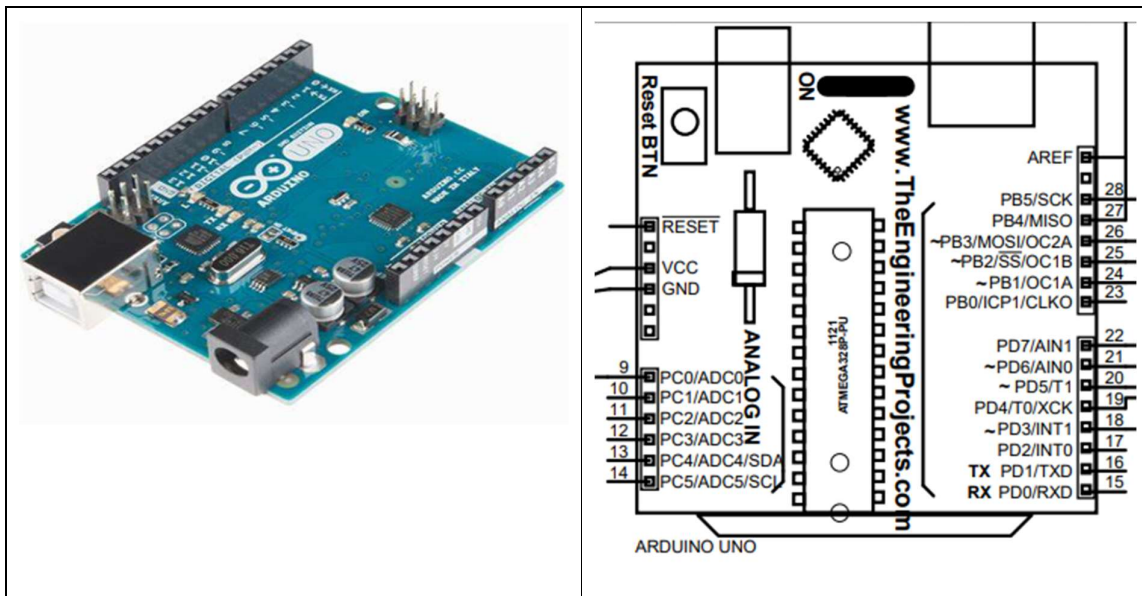
Bảng 2.4: Bảng các thông số chân module TM1637

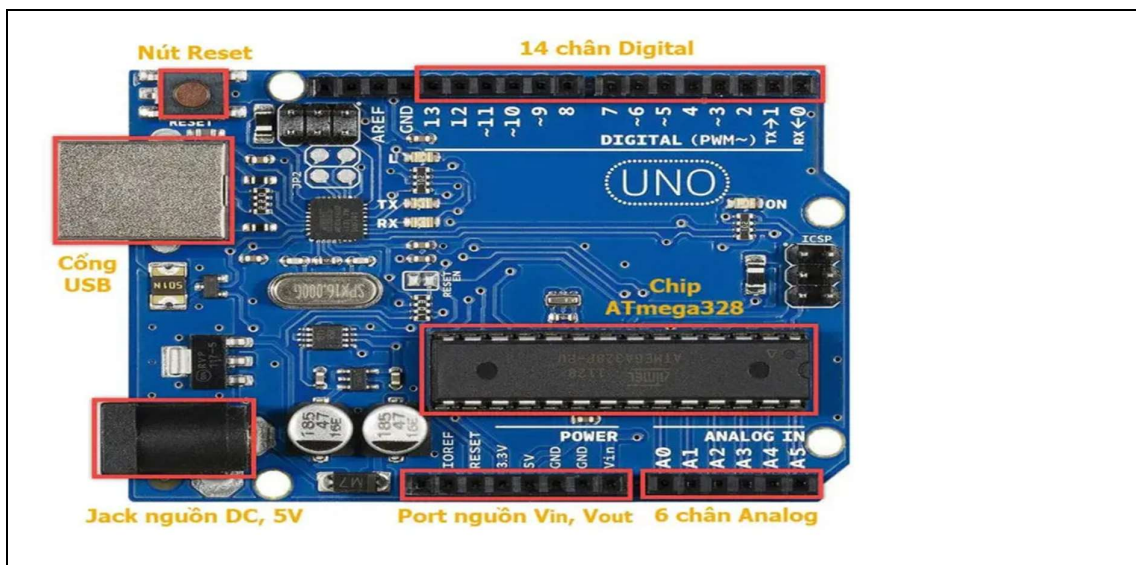
2.5 Vi điều khiển Arduino

Arduino là một board mạch vi xử lý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, hoặc ARM Atmel 32-bit. Arduino là một nền tảng mã nguồn mở, nguyên mẫu và tính đơn giản của nó làm cho nó trở thành lý tưởng cho những người có sở thích sử dụng cũng như các chuyên gia.

Hiện nay trên thị trường đã có nhiều mẫu Arduino với đa dạng chức năng và kích thước. Tùy vào mục đích sử dụng mà ta có thể lựa chọn các loại vi điều khiển khác nhau, có thể kể đến 1 số mạch tiêu biểu như Arduino MEGA 2650, Arduino Nano, ... trong đó lựa chọn phổ thông nhất chính là Arduino Uno

2.5.1 Arduino Uno R3





Hình 2.5-1: Bo mạch Arduino Uno R3

Arduino Uno R3 là một bảng mạch vi điều khiển nguồn mở dựa trên vi điều khiển Microchip ATmega328 được phát triển bởi Arduino.cc. Arduino là một nền tảng mã nguồn mở, nguyên mẫu và tính đơn giản của nó làm cho nó trở thành lý tưởng cho những người có sở thích sử dụng cũng như các chuyên gia. Arduino Uno có 14 chân đầu vào / đầu ra kỹ thuật số (trong đó 6 có thể được sử dụng làm đầu ra PWM), 6 đầu vào tương tự, bộ tạo dao động 16 MHz, kết nối USB, giắc cắm nguồn, tiêu đề ICSP và nút đặt lại. Nó chứa mọi thứ cần thiết để hỗ trợ vi điều khiển; chỉ cần kết nối nó với một máy tính bằng cáp USB hoặc cấp điện cho nó bằng bộ chuyển đổi AC-to-DC hoặc pin để bắt đầu.

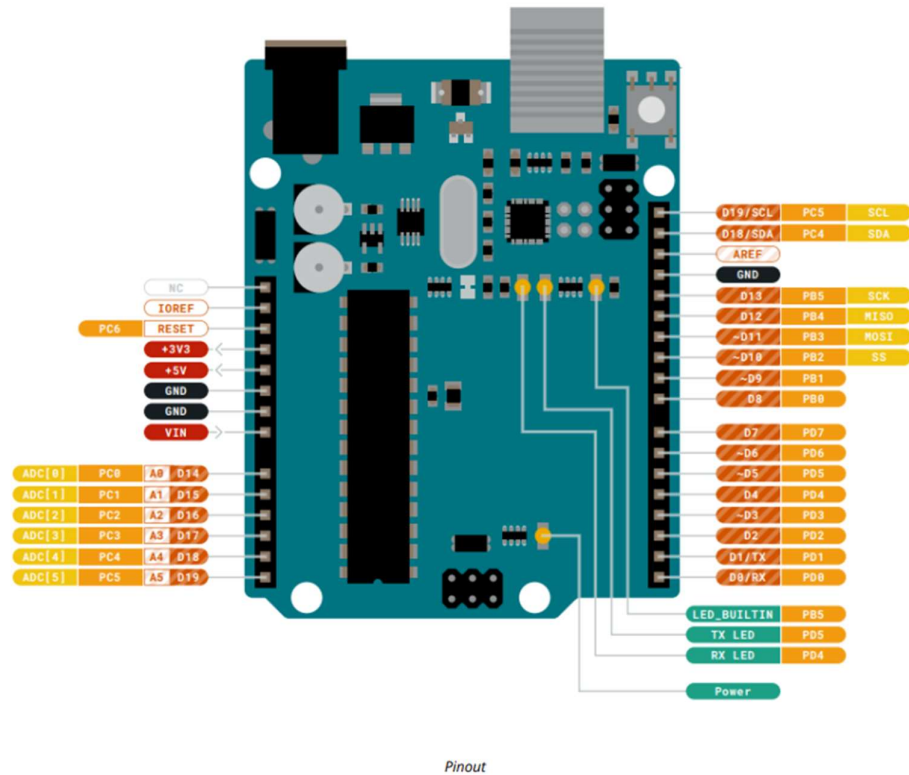
- **Các thông số cơ bản**

Vi điều khiển	ATmega328 (8bit)
Điện áp hoạt động	5V DC (chỉ được cấp qua cổng USB)
Tần số hoạt động	16 MHz
Dòng tiêu thụ	khoảng 40mA

Điện áp vào khuyến dùng	7-12V DC
Điện áp vào giới hạn	6-20V DC
Số chân Digital I/O	14 (có 6 chân 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 (3.3V)	50 mA
Bộ nhớ flash	32 KB (ATmega328) với 0.5KB dùng bởi bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)

Bảng 2.5-1: Thông số của vi điều khiển Arduino Uno R3

- **Các chân GPIO:**



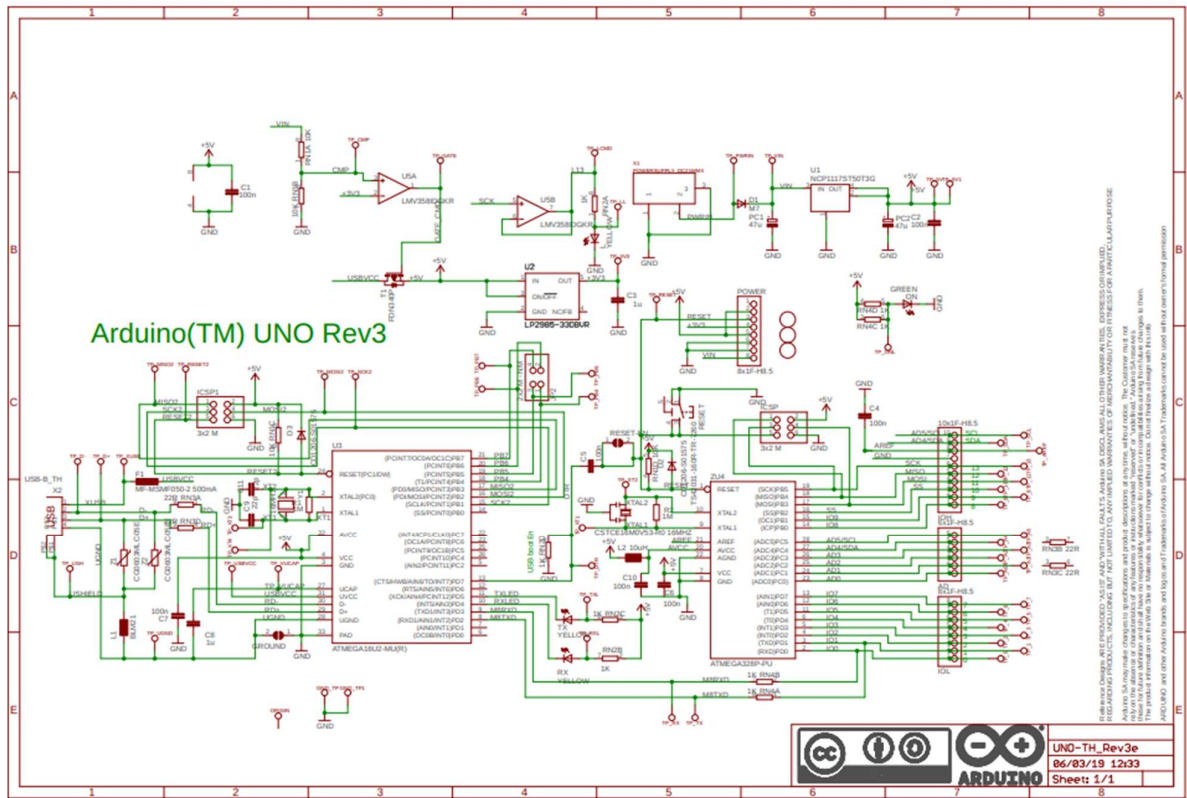
Pin	Function	Type	Description
1	D0	Digital/GPIO	Digital pin 0/GPIO
2	D1	Digital/GPIO	Digital pin 1/GPIO
3	D2	Digital/GPIO	Digital pin 2/GPIO
4	D3	Digital/GPIO	Digital pin 3/GPIO
5	D4	Digital/GPIO	Digital pin 4/GPIO
6	D5	Digital/GPIO	Digital pin 5/GPIO
7	D6	Digital/GPIO	Digital pin 6/GPIO
8	D7	Digital/GPIO	Digital pin 7/GPIO
9	D8	Digital/GPIO	Digital pin 8/GPIO
10	D9	Digital/GPIO	Digital pin 9/GPIO
11	SS	Digital	SPI Chip Select
12	MOSI	Digital	SPI1 Main Out Secondary In
13	MISO	Digital	SPI Main In Secondary Out
14	SCK	Digital	SPI serial clock output
15	GND	Power	Ground
16	AREF	Digital	Analog reference voltage
17	A4/SD4	Digital	Analog input 4/I2C Data line (duplicated)
18	A5/SD5	Digital	Analog input 5/I2C Clock line (duplicated)

Bảng 2.5-2: thông tin các chân analog của Arduino Uno R3

Pin	Function	Type	Description
1	NC	NC	Not connected
2	IREF	IREF	Reference for digital logic V - connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	+5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage Input
9	A0	Analog/GPIO	Analog input 0 /GPIO
10	A1	Analog/GPIO	Analog input 1 /GPIO
11	A2	Analog/GPIO	Analog input 2 /GPIO
12	A3	Analog/GPIO	Analog input 3 /GPIO
13	A4/SDA	Analog input/I2C	Analog input 4/I2C Data line
14	A5/SCL	Analog input/I2C	Analog input 5/I2C Clock line

Bảng 2.5-3: thông tin các chân analog và các port nguồn của Arduino Uno R3

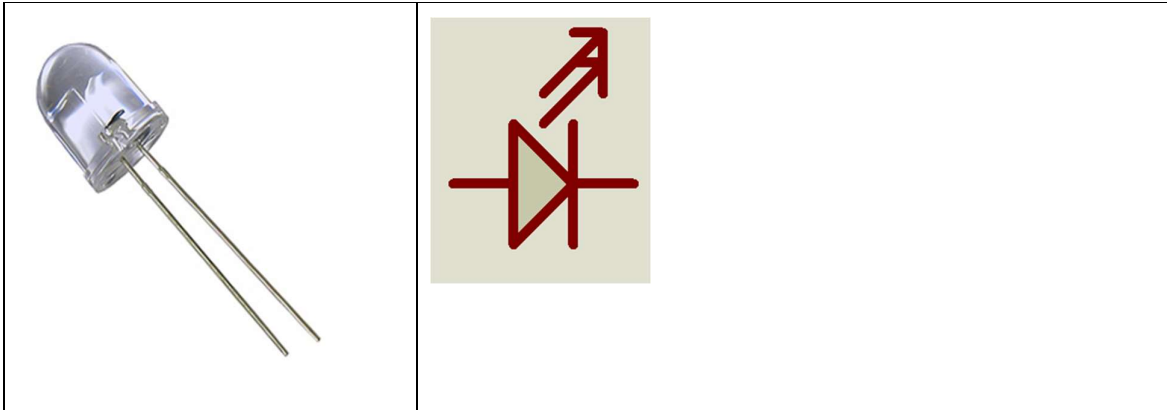
Sơ đồ nguyên lý của Arduino Uno R3



Hình 2.5-3: Sơ đồ nguyên lý của Arduino Uno R3

[UNO-TH_Rev3e.sch \(arduino.cc\)](https://arduino.cc/UNO-TH_Rev3e.sch)

2.6 Led đơn



Hình 2.6: Hình ảnh LED đơn

LED (Light Emitting Diode) là linh kiện cơ bản trong điện tử, dễ sử dụng và có nhiều ứng dụng trong thực tế. Nguyên lý hoạt động của led đơn là Led sẽ phát sáng khi dòng điện đi từ cực dương (anode) sang cực âm (cathode), làm các electron và lỗ trống kết hợp tại vùng tiếp giáp, phát ra ánh sáng.

Các thông số của Led đơn như sau:

- Điện áp hoạt động Thường từ 1.8V đến 3.3V tùy loại LED
- Dòng điện hoạt động Thường từ 10mA đến 20mA, tùy loại và mục đích sử dụng.
- Tuổi thọ: Cao, thường từ 50,000 đến 100,000 giờ.

2.7 Buzzer



Hình 2.7: Hình ảnh Buzzer

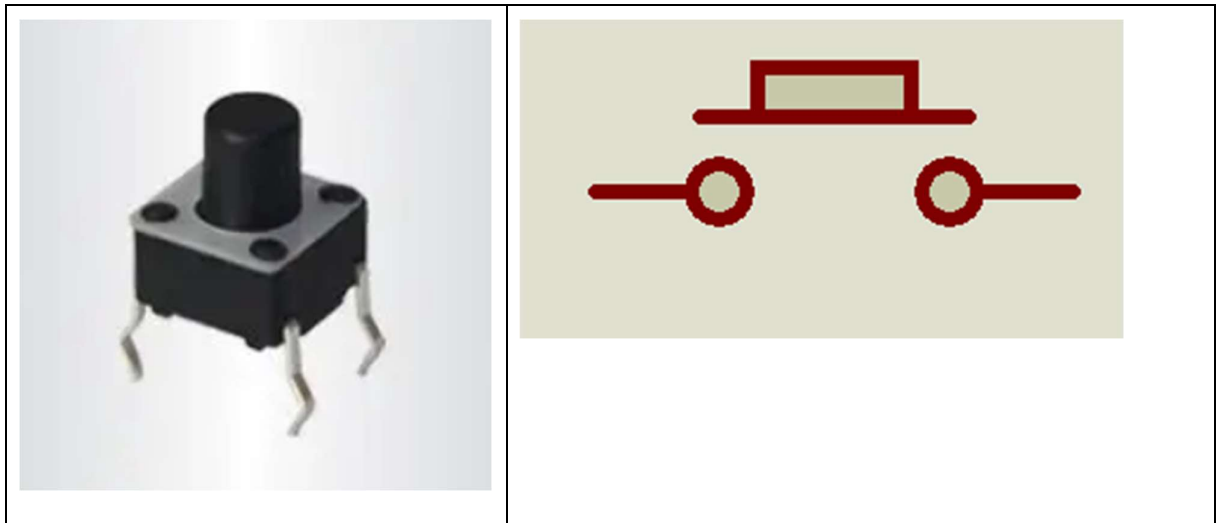
Buzzer (còi báo) là linh kiện cơ bản và dễ sử dụng trong điện tử. Nguyên lý hoạt động của Buzzer là chuyển đổi tín hiệu điện thành âm thanh. Khi có dòng điện chạy qua, tấm kim loại bên trong rung động tạo ra âm thanh. Buzzer có 2 loại là :

- **Active Buzzer:** Tự tạo ra âm thanh khi được cấp nguồn, không cần tín hiệu điều khiển.
- **Passive Buzzer:** Cần tín hiệu dao động (PWM) để phát ra âm thanh.

Thông số kĩ thuật của Buzzer:

- Điện áp hoạt động: từ 3V đến 12V.
- Dòng điện hoạt động: g từ 10mA đến 30mA.
- Tần số âm thanh: Khoảng 1kHz đến 10kHz
- Độ lớn âm thanh: Có thể đạt tới 85dB hoặc cao hơn, tùy thuộc vào loại và thiết kế.

2.8 Nút nhấn



Hình 2.8 : Hình ảnh thực tế của nút nhấn

Nút nhấn (push button) là một thiết bị cơ khí đơn giản được sử dụng để tạo ra kết nối điện tạm thời khi được nhấn xuống và ngắt kết nối khi thả ra. Nguyên lý hoạt động của nút nhấn là khi nhấn nút, các tiếp điểm bên trong nút kết nối lại với nhau, tạo ra một mạch điện kín. Khi thả nút ra, các tiếp điểm tách ra và ngắt mạch. Từ đó sẽ đổi trạng thái của hệ thống. Có 2 loại nút nhấn là nút nhấn tạm thời và nút nhấn duy trì.

- Nút nhấn tạm thời (Momentary Push Button): Kết nối chỉ duy trì khi nút đang được nhấn. Khi thả ra, nút trở lại trạng thái ban đầu (hở mạch).
- Nút nhấn duy trì (Latching Push Button): Khi nhấn lần đầu, nút giữ kết nối (đóng mạch). Khi nhấn lần thứ hai, nút ngắt kết nối (hở mạch).

Đặc điểm chung của nút nhấn:

- Điện áp và dòng điện: Phụ thuộc vào thiết kế của nút, nhưng thường nút nhấn được thiết kế để xử lý các tín hiệu điện áp thấp (3V - 12V) và dòng điện nhỏ (ít hơn 1A).
- Chân kết nối: Thường có 2 hoặc 4 chân, trong đó 2 chân chính để nối mạch và 2 chân phụ dùng cho các kết nối khác như LED báo.

Chương 3: TÍNH TOÁN VÀ THIẾT KẾ HỆ THỐNG

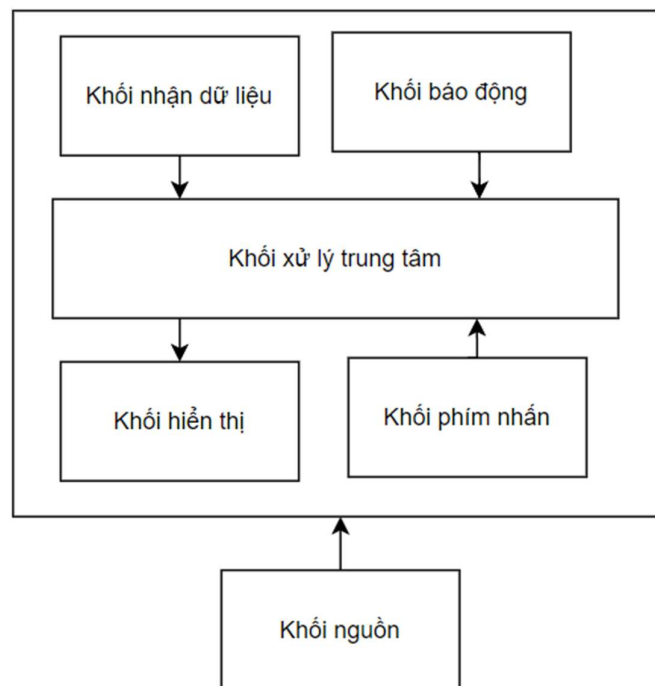
3.1 Yêu cầu và sơ đồ khối của hệ thống

3.1.1 Yêu cầu của hệ thống

Hệ thống này sẽ gồm các chức năng như sau:

- Thu thập giá trị nồng độ cồn
- Hiển thị giá trị nồng độ cồn
- Có nút chuyển đổi trạng thái
- Báo động khi vượt ngưỡng cho phép

3.1.2 Sơ đồ khối của hệ thống



Hình 3.1.2-1 Sơ đồ khối của hệ thống

- Chức năng từng khối:

- **Khối Xử lý trung tâm:** Khối này sẽ tính toán, xử lý các dữ liệu, yêu cầu từ các khối hiển thị, khối nút nhấn, khối nhận dữ liệu, khối báo động.
- **Khối hiển thị:** Khối này sẽ hiển thị các dữ liệu đã được xử lý thông qua khối xử lý trung tâm

- **Khối nhận dữ liệu:** Khối này có chức năng lấy thông tin từ môi trường sau đó sẽ chuyển về khối xử lý
- **Khối phím nhấn:** có chức năng điều khiển, chuyển đổi hệ thống
- **Khối báo động:** có chức năng báo động khi dữ liệu vượt ngưỡng giá trị cho phép
- **Khối nguồn:** có chức năng cung cấp nguồn cho hệ thống hoạt động.
- **Giải thích hoạt động hệ thống:** Đầu tiên khối nhận dữ liệu sẽ bắt đầu lấy dữ liệu từ bên ngoài môi trường, sau đó đưa cho khối xử lý trung tâm làm nhiệm vụ phân tích và tính toán dữ liệu đó. Rồi tiếp theo sẽ đẩy dữ liệu đó lên khối hiển thị để hiển thị kết quả, đồng thời nếu dữ liệu chuyển đến trạng thái đã được cài đặt thì sẽ có sự tham gia của khối báo động. Khối nút nhấn cũng sẽ được tác động vào hệ thống nhằm để thay đổi trạng thái. Trong suốt quá trình này, khối nguồn sẽ cấp nguồn cho toàn bộ hệ thống hoạt động.

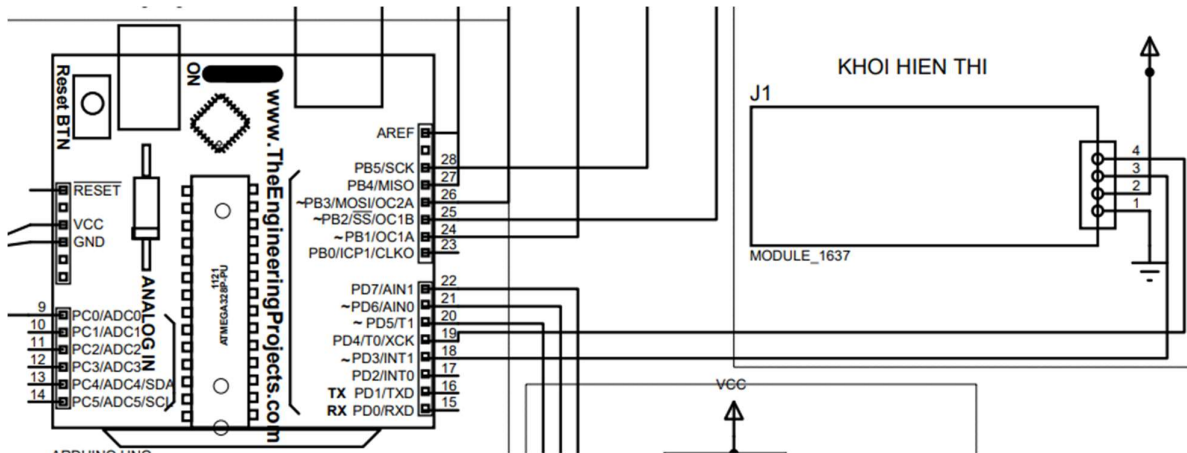
3.2 Thiết kế phần cứng hệ thống

3.2.1 Khối nhận dữ liệu

Trên thị trường hiện nay cũng có nhiều loại cảm biến nồng độ cồn, chẳng hạn như MQ-3, MQ-135, MQ-6, AT-6000,... nhưng ở đây em chọn MQ-3 là vì cảm biến này phổ biến nhất trong các loại cảm biến kể trên, ngoài ra cảm biến này cũng dễ sử dụng và giá thành rẻ, phù hợp với dự án. Và trên thị trường hiện nay có tích hợp sẵn module nồng độ cồn MQ-3 khiến cho việc đo nồng độ cồn sẽ trở nên thuận tiện và dễ dàng hơn.

Module nồng độ cồn MQ-3 có 4 chân kết nối bao gồm VCC, GND, D0 và A0. Các kết nối được thực hiện như sau:

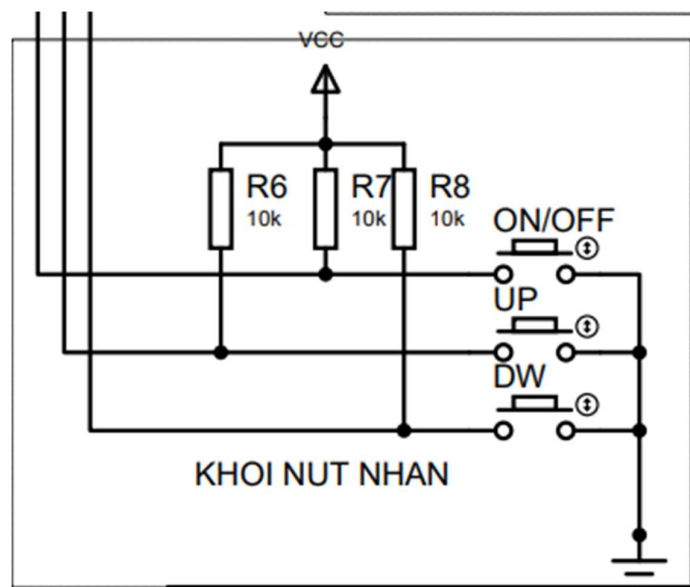
- Chân VCC của module sẽ kết nối với chân VCC có trong Arduino Uno R3.
- Chân GND sẽ nối xuống GND.
- Chân A0 là chân tín hiệu tương tự, chân này sẽ nối đến chân A0 của Arduino Uno R3.
- Chân D0 không nối vì đây là chân xuất ra giá trị số (giá trị 0 và 1), điều này chỉ cho biết môi trường có cồn hay không chứ không có giá trị cụ thể như chân A0.



Hình3.2.2: sơ đồ nguyên lý nối chân của Module_TM1637

3.2.3 Khối nút nhấn

Khối nút nhấn dùng để đổi trạng thái của hệ thống. Ở đây em sẽ sử dụng 3 nút nhấn tích cực mức thấp, được dùng trở nội có sẵn trong Arduino Uno R3 để điều khiển. 3 nút nhấn này sẽ được nối với chân 6 và 7 của Arduino Uno R3. 1 nút nhấn dùng để bật/ tắt hệ thống, 1 nút nhấn dùng để tăng ngưỡng và nút còn lại dùng giảm ngưỡng nồng độ cồn.



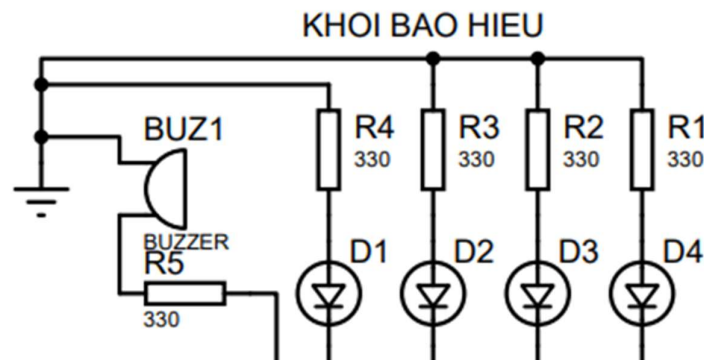
Hình 3.2.3: Sơ đồ nguyên lý nối chân của khối nút nhấn

3.2.4 Khối báo động

Khối báo động dùng để báo động khi nồng độ cồn vượt ngưỡng hoặc là có chuyển đổi trạng thái từ nút nhấn. Để thể hiện được khối báo động này thì em sẽ dùng các linh

kiện cơ bản như đèn LED và Buzzer để báo động cho hệ thống. Ở đây em sẽ dùng 4 đèn Led với màu khác nhau để phân biệt được các trạng thái của hệ thống. Khi không có cồn thì Led vàng sẽ sáng, khi có cồn thì led đỏ sẽ sáng. Còn khi thay đổi giá trị ngưỡng của hệ thống thì Led xanh lá và xanh dương sẽ sáng.

- Buzzer sẽ được nối vào chân 12 của Arduino.
- 4 Led sẽ lần lượt nối vào chân 9,10,11,13 của Arduino.



Hình 3.2.4: sơ đồ nguyên lý của khối báo động

3.2.5 Khối xử lý trung tâm

Hiện nay trên thị trường có rất nhiều loại board Arduino như: Mega, Uno, Nano,...với nhiều tính năng, kích cỡ và giá thành khác nhau. Em chọn board arduino Uno R3 sử dụng làm khối xử lý trung tâm bởi vì đây là vi điều khiển có kích thước vừa đủ, có đủ các port để kết nối với các khối còn lại.

Khối xử lý trung tâm sẽ được kết nối như sau:

- Chân A0 sẽ nối với chân A0 của module nồng độ cồn MQ-3.
- Chân 5, 6 và chân 7 sẽ nối đến 3 nút nhấn của khối nút nhấn.
- Chân 3 và chân 4 sẽ lần lượt nối đến chân DIO và chân CLK của module hiển thị led 7 đoạn TM1637.
- Chân 9, 10, 11,13 sẽ nối đến 4 led đơn trong khối báo động.
- Chân 12 sẽ nối đến buzzer trong khối báo động.

Qua các mô tả trên ta có được sơ đồ kết nối toàn mạch

3.2.6 Khối nguồn

Để thiết kế khối nguồn, ta tính toán dòng tiêu thụ và điện áp của các thiết bị có trong hệ thống. Sau đây là bảng liệt kê dòng và áp của các linh kiện:

STT	Thiết bị	Dòng tiêu thụ	Điện áp
1	Module nồng độ cồn MQ-3	150 mA	5 V
2	LED	20 mA	1.8V- 3.3V
3	Buzzer	30 mA	5V
4	Module TM1637	80 mA	5 V
5	Arduino Uno R3	40 mA	5 V

Bảng 3.2.6: Bảng tính toán nguồn hoạt động của hệ thống

Do sử dụng 4 Led nên dòng tiêu thụ của LED : $I_{LED} = 20 \times 4 = 80 \text{mA}$

Từ đó cho thấy tổng dòng điện cho toàn

mạch là $I = I_{MQ-3} + I_{Led} + I_{Buzzer} + I_{TM1637} + I_{Uno}$

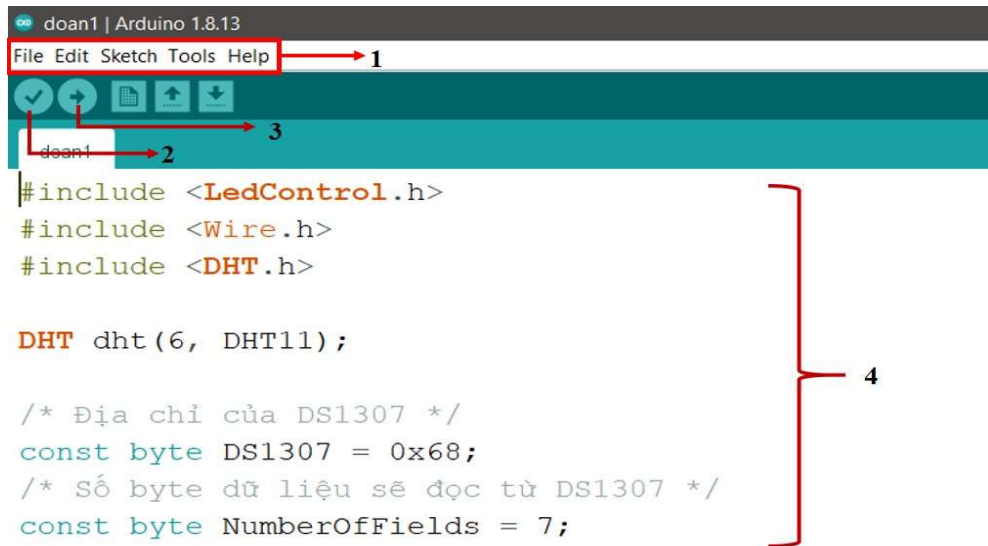
$$= 0.15 + 0.08 + 0.03 + 0.08 + 0.04 = 0.38 \text{ (A)}$$

Suy ra ở hệ thống này cần nguồn 5V- 0.38A để hoạt động.

Để tối ưu hóa cho hệ thống nên em sẽ chọn Pin 9V để cấp nguồn cho hệ thống

3.3 Thiết kế phần mềm hệ thống

Trong đề tài em sử dụng phần mềm Arduino IDE để lập trình và giải quyết các vấn đề.

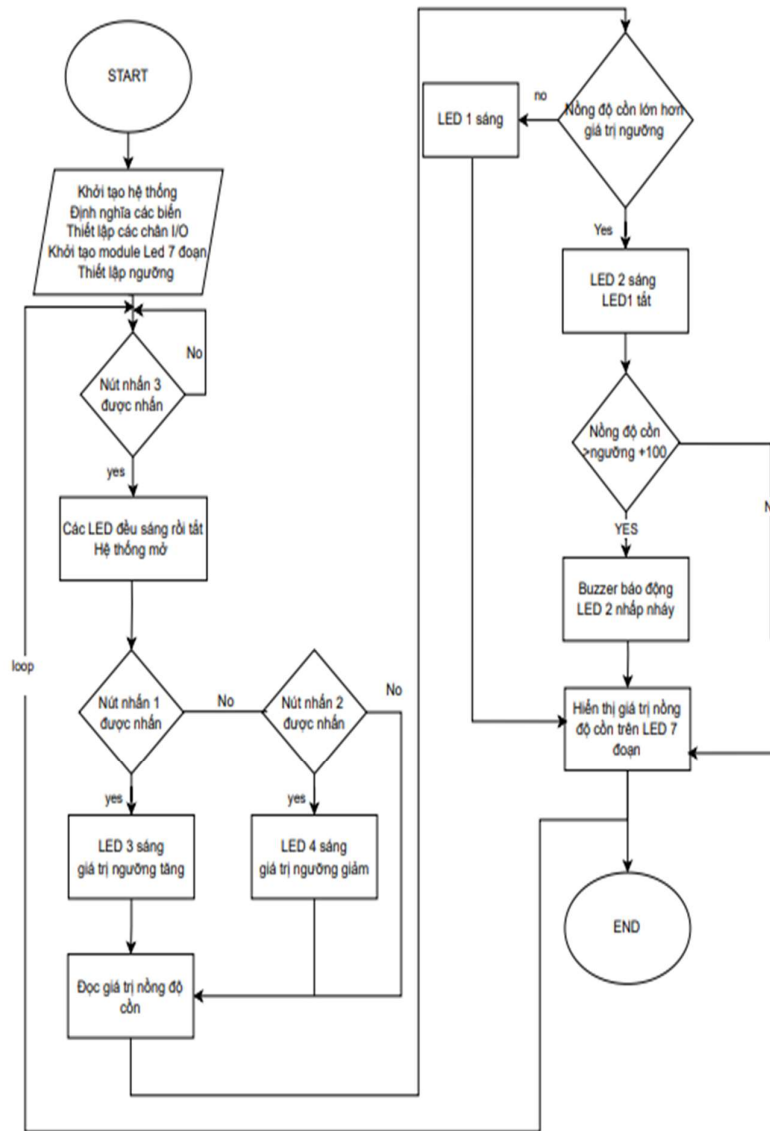


Hình 3.3: Giao diện phần mềm Arduino IDE

Giao diện của phần mềm Arduino IDE có nhiều phần, tuy nhiên chúng ta chú ý đến những phần quan trọng như được nêu ra trong hình trên. Chức năng của từng phần như sau:

- Vùng 1: Menu lệnh: Dùng để thêm thư viện, lưu, tạo project mới,....
- Vùng 2: Nút kiểm tra chương trình (built): Dùng để kiểm tra xem chương trình được viết có lỗi không. Nếu chương trình bị lỗi thì phần mềm Arduino IDE sẽ hiển thị thông tin lỗi ở vùng thông báo thông tin.
- Vùng 3: Nút nạp chương trình xuống board Arduino: Dùng để nạp chương trình được viết xuống mạch Arduino. Trong quá trình nạp, chương trình sẽ được kiểm tra lỗi trước sau đó mới thực hiện nạp xuống mạch Arduino.
- Vùng 4: Vùng lập trình: Vùng này để người lập trình thực hiện việc lập trình cho chương trình của mình.

3.4 Lưu đồ giải thuật của hệ thống



Hình 3.4: Lưu đồ giải thuật của hệ thống

Giải thích lưu đồ:

- Đầu tiên ta sẽ khởi tạo các biến, các chân I/O và giá trị ngưỡng ban đầu.

- Kiểm tra nút nhấn 1: nếu nút nhấn 1 được nhấn thì hệ thống bật/tắt, đồng thời các led sẽ sáng rồi tắt báo hiệu hệ thống mở.

- Kiểm tra nút nhấn 2: nếu nút nhấn 2 được nhấn thì hệ thống sẽ tăng ngưỡng đã được thiết lập trước đó

Kiểm tra nút nhấn 3: nếu nút nhấn 3 được nhấn thì hệ thống sẽ giảm ngưỡng đã được thiết lập trước đó

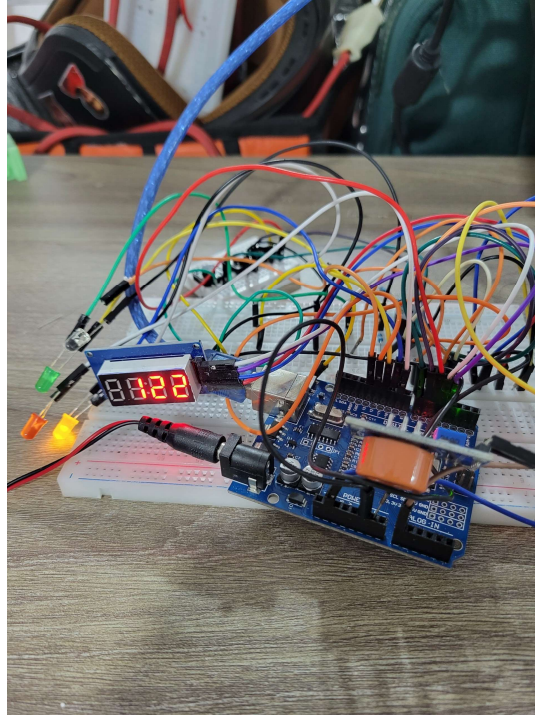
- Tiến hành tính toán đọc từ cảm biến lấy giá trị nồng độ cồn và so sánh với ngưỡng đã được thiết lập. Nếu nồng độ cồn lớn hơn ngưỡng thì LED 2 sẽ sáng. Còn nếu nồng độ cồn lớn hơn ngưỡng cho phép thì LED 2 sẽ nhấp nháy và buzzer sẽ bật. Nếu nồng độ cồn không vượt qua ngưỡng cho phép thì LED 1 sẽ sáng

- Cuối cùng hiển thị giá trị nồng độ cồn lên module led 7 đoạn

Chương 4: KẾT QUẢ_NHẬN XÉT_ĐÁNH GIÁ

4.1 Kết quả sản phẩm

Qua quá trình thiết kế phần cứng, chọn lựa linh kiện, em đã tiến hành kiểm tra các kết nối các module của các khối và cho ra kết quả như các hình bên dưới.



Hình 4.1 Test module sản phẩm

Trong quá trình test sản phẩm trên testboard thì em nhận thấy hệ thống hoạt động tốt, đúng với yêu cầu đã đề ra trước đó.

4.2 Kết quả toàn hệ thống hoạt động

Sau khi kết nối hệ thống ổn định, em tiến hành hoàn thiện hệ thống thành mô hình từ các module rời và tạo thành sản phẩm như hình bên dưới



Hình 4.2 Mô hình sản phẩm

Hoạt động của sản phẩm:

- Khi chúng ta ấn phím đỏ thì hệ thống sẽ mở, đồng thời 4 led đơn và Led 7 đoạn sẽ sáng lên, sau đó 4 led sẽ tắt. Khi nhấn 1 lần nữa thì hệ thống sẽ tắt, 4 Led đơn sáng rồi tắt và Led 7 đoạn tắt.
- Khi ấn phím xanh lá thì hệ thống sẽ tăng ngưỡng nồng độ cồn lên 1 khoảng 50 ppm. Lúc này cảm biến đọc được sẽ so sánh với ngưỡng đó và nếu nồng độ cồn nhỏ hơn ngưỡng này thì Led vàng sẽ sáng, còn nếu lớn hơn ngưỡng này thì Led đỏ sáng, Led vàng tắt. Khi mà nồng độ cồn lớn hơn ngưỡng 100ppm thì lúc này Led đỏ sẽ chớp tắt liên tục và buzzer sẽ kêu.
- Khi ấn phím xanh dương thì hệ thống sẽ giảm ngưỡng nồng độ cồn lên 1 khoảng 50 ppm. Lúc này cảm biến đọc được sẽ so sánh với ngưỡng đó và nếu nồng độ cồn nhỏ hơn ngưỡng này thì Led vàng sẽ sáng, còn nếu lớn hơn ngưỡng này thì Led đỏ sáng, Led vàng tắt. Khi mà nồng độ cồn lớn hơn ngưỡng 100ppm thì lúc này Led đỏ sẽ chớp tắt liên tục và buzzer sẽ kêu.

Chương 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

Với sự hỗ trợ tận tình từ giáo viên hướng dẫn về cơ bản đã hoàn thành được những mục tiêu đề ra là thiết kế mạch đo nồng độ cồn hiển thị trên Led 7 đoạn. Trong quá trình thực hiện, tuy gặp không ít khó khăn từ phần cứng đến phần mềm. Mặc dù gặp không ít khó khăn như thế nhưng em đã cố gắng nghiên cứu vượt qua và tích lũy cho mình một số kinh nghiệm mới, kiến thức mới để hoàn thành đề tài.

5.2 Hạn chế và hướng phát triển

5.2.1 Hạn chế

- Chi phí phần cứng khá cao.
- Còn ảnh hưởng nhiều bởi nhiễu.
- Chỉ chạy được trên môi trường lý tưởng.
- Phần cứng tự tạo nên chưa đạt độ chính xác cao.
- Nút nhấn thỉnh thoảng vẫn chưa đổi trạng thái mặc dù đã tối ưu hóa.
- Cảm biến cần phải hoạt động trước 1 khoảng thời gian thì mới có thể lấy dữ liệu đúng được
- Sản phẩm vẫn còn kích thước lớn, chưa thuận tiện lắm ở mặt di động.

5.2.2 Hướng phát triển

Sau khi hoàn thiện xong mô hình, em có những hướng phát triển như sau:

- Cải thiện thêm thuật toán để củng cố mô hình hoàn thiện hơn.
- Hiển thị giá trị trên linh kiện có hình ảnh bắt mắt hơn.

- Sử dụng cảm biến có độ chính xác nhanh hơn, cao hơn nhằm có được giá trị nồng độ còn đúng hơn.
- Thiết kế sản phẩm có kích thước nhỏ gọn hơn, bắt mắt hơn.

TÀI LIỆU THAM KHẢO

[1] UNO R3 | Arduino Documentation

[2] <https://docs.arduino.cc/hardware/uno-rev3/>

[3] <https://ledhd.vn/tin-tuc/led-7-doan-la-gi.html>

[4] <https://youtu.be/AOipuKb0JAw?si=G6LulA0EMlf6YhsL>

[5] <https://youtu.be/2CMThDM0wfY?si=pdUFkg0Fy22IDxad>

[6] <https://dientuviet.com/giao-tiep-led-7-doan-voi-arduino/>

[7] Datasheet MQ3:

TM1637 Datasheet(PDF) - Shenzhen Titan Micro Electronics Co., Ltd. (alldatasheet.com)

[8] Datasheet Arduino:

<https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf>

[9] Datasheet module TM1637:

Datasheet-tm1637-based-4-bits-red-digital-tube-led-display-module.pdf (robu.in)

PHỤ LỤC

```
#include <TM1637Display.h>
```

```
#include <Arduino.h>
```

```
#define CLK 4
```

```
#define DIO 3
```

```
#define LED_1 9
```

```
#define LED_2 10
```

```
#define LED_3 11
```

```
#define LED_4 13
```

```
#define buzzer 12
```

```
const int buttonPin_1 = 5;
```

```
const int buttonPin_2 = 6;
```

```
const int buttonPin_3 = 7;
```

```
float value;
```

```
float threshold = 0;
```

```
int sensorPin = A0; // Cảm biến MQ-3 kết nối với chân A0
```

```
TM1637Display display(CLK, DIO);
```

```
boolean buttonState1 = HIGH;
```

```
boolean buttonState2 = HIGH;
```

```
boolean buttonState3 = HIGH;
```

```
boolean systemOn = true; // State variable to keep track of system status
```

```
void setup() {
```

```
    Serial.begin(9600);
```

```
    pinMode(LED_1, OUTPUT);
```

```
    pinMode(LED_2, OUTPUT);
```

```
    pinMode(LED_3, OUTPUT);
```

```
    pinMode(LED_4, OUTPUT);
```

```
    pinMode(buzzer, OUTPUT);
```

```
    pinMode(buttonPin_1, INPUT_PULLUP); // New button for turning the system  
on/off
```

```
    pinMode(buttonPin_2, INPUT_PULLUP); // Kết nối nút nhấn với điện trở kéo lên  
bên ngoài
```

```
    pinMode(buttonPin_3, INPUT_PULLUP); // Kết nối nút nhấn với điện trở kéo lên  
bên ngoài
```

```
    threshold = 350;
```

```
    display.setBrightness(0x0f); // Đặt độ sáng tối đa
```

```
    display.clear();
```

```
    delay(2000);
```

```
}
```

```
boolean debounceButton(int buttonPin, boolean state) {
```

```

boolean stateNow = digitalRead(buttonPin);

if (state != stateNow) {

    delay(100); // Chờ 100ms để debounce

    stateNow = digitalRead(buttonPin);

}

return stateNow;

}

void button() {

    buttonState1 = debounceButton(buttonPin_1, buttonState1);

    if (buttonState1 == LOW) {

        systemOn = !systemOn; // Toggle system state

        digitalWrite(LED_1, HIGH); // Briefly light up LED 3 as feedback
        digitalWrite(LED_2, HIGH);
        digitalWrite(LED_3, HIGH);
        digitalWrite(LED_4, HIGH);

        delay(500);

        digitalWrite(LED_1, LOW); // Briefly light up LED 3 as feedback
        digitalWrite(LED_2, LOW);
        digitalWrite(LED_3, LOW);
        digitalWrite(LED_4, LOW);

        Serial.println("nut 1 da duoc nhan ");

    }

    if (!systemOn) {

```



```
    return; // If system is off, skip the rest of the button checks  
}
```

```
buttonState2 = debounceButton(buttonPin_2, buttonState2);  
if (buttonState2 == LOW) {  
    threshold += 50;  
    digitalWrite(LED_3, HIGH);  
    delay(500); // Đèn LED sáng trong 500ms  
    digitalWrite(LED_3, LOW);  
    Serial.println("nut 2 da duoc nhan ");  
}
```

```
buttonState3 = debounceButton(buttonPin_3, buttonState3);  
if (buttonState3 == LOW) {  
    threshold -= 50;  
    digitalWrite(LED_4, HIGH);  
    delay(500); // Đèn LED sáng trong 500ms  
    digitalWrite(LED_4, LOW);  
    Serial.println("nut 3 da duoc nhan ");  
}
```

```
}
```

```
void loop() {  
  
    button();  
  
    if (!systemOn) {  
        display.clear(); // Clear the display if the system is off  
        digitalWrite(LED_1, LOW);  
        digitalWrite(LED_2, LOW);  
        digitalWrite(LED_3, LOW);  
        digitalWrite(LED_4, LOW);  
        digitalWrite(buzzer, LOW);  
        return;  
    }  
  
    int rawValue = analogRead(sensorPin);  
    value = rawValue / 1024.0 * 5.0; // Chuyển đổi giá trị analog sang điện áp (0-5V)  
  
    float alcoholValue = (value / 5.0) * 500.0; // Điều chỉnh công thức tính toán  
  
    Serial.print("Alcohol Value: ");  
    Serial.println(alcoholValue);  
  
    display.showNumberDec(alcoholValue, false, 4, 0); // Hiển thị giá trị nồng độ cồn  
  
    if (alcoholValue < threshold) {
```

```
    digitalWrite(LED_2, HIGH); // Bật LED 2
    digitalWrite(LED_1, LOW); // Tắt LED 1
}
// Khi có cồn
else {
    digitalWrite(LED_2, LOW); // Tắt LED 2
    digitalWrite(LED_1, HIGH); // Bật LED 1
}

if (alcoholValue > (threshold + 100)) {
    digitalWrite(buzzer, HIGH);
    for (int i = 0; i < 20; i++) {
        digitalWrite(LED_1, HIGH);
        delay(100);
        digitalWrite(LED_1, LOW);
        delay(100);
    }
} else {
    digitalWrite(buzzer, LOW);
}

delay(1000); // Thêm một khoảng trễ ngắn để tránh vòng lặp quá nhanh
}
```