

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**



ĐỒ ÁN TỐT NGHIỆP

NGÀNH ĐIỆN TỬ ĐỘNG CÔNG NGHIỆP

Sinh viên : Kim Văn Dinh

Giảng viên hướng dẫn: TS. Đoàn Hữu Chúc

Hải Phòng -2023

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**



**THIẾT KẾ HỆ THỐNG ĐO
VÀ GIÁM SÁT ĐỘ ẨM ĐẤT**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY
NGÀNH ĐIỆN TỬ ĐỘNG CÔNG NGHIỆP**

Sinh viên thực hiện : Kim Văn Dinh

Giảng viên hướng dẫn: TS. Đoàn Hữu Chức

Hải Phòng - 2023

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG

NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên : Kim Văn Dinh - **MSV** : 1412102101

Lớp : DC 1901

Ngành : Điện Tự Động Công Nghiệp

Tên đề tài : Nghiên cứu, thiết kế hệ thống đo và giám sát
nồng độ chất độc – hại trong không khí
ứng dụng công nghệ Internet vạn vật

NHIỆM VỤ ĐỀ TÀI

1.Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp (về lý luận, thực tiễn, các số liệu cần tính toán và các bản vẽ).

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

2. Các số liệu cần thiết để tính toán.

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

3.Địa điểm thực tập tốt nghiệp.

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

CÁC CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Họ và tên : Đoàn Hữu Chức

Học hàm, học vị : Tiến sĩ

Cơ quan công tác : Trường Đại học quản lý và công nghệ Hải Phòng

Nội dung hướng dẫn:

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

Đề tài tốt nghiệp được giao ngày 07 tháng 11 năm 2022

Yêu cầu phải hoàn thành xong trước ngày 18 tháng 02 năm 2023

Đã nhận nhiệm vụ ĐTTN

Sinh viên

Đã giao nhiệm vụ ĐTTN

Giảng viên hướng dẫn

Kim Văn Dinh

Đoàn Hữu Chức

Hải Phòng, ngày tháng năm 2023

TRƯỞNG KHOA

TS. Đoàn Hữu Chức

Cộng Hòa Xã Hội Chủ Nghĩa Việt Nam
Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên : Đoàn Hữu Chức

Đơn vị công tác : Trường Đại học Quản lý và Công nghệ Hải Phòng

Họ và tên sinh viên : Kim Văn Dinh

Chuyên ngành : Điện Tự Động Công Nghiệp

Nội dung hướng dẫn : Toàn bộ đề tài

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

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

2. Đánh giá chất lượng của đồ án/khóa luận (so với nội dung yêu cầu đã đề ra trong nhiệm vụ Đ.T.T.N, trên các mặt lý luận, thực tiễn, tính toán số liệu...)

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

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Được bảo vệ Không được bảo vệ Điểm hướng dẫn

Hải Phòng, ngày.....tháng.....năm 2023

Giảng viên hướng dẫn

(ký và ghi rõ họ tên)

PHIẾU NHẬN XÉT CỦA GIÁNG VIÊN CHẤM PHẢN BIỆN

Họ và tên giảng viên

Đơn vị công tác:.....

Họ và tên sinh viên: Chuyên ngành:.....

Đề tài tốt nghiệp:

.....

1. Phần nhận xét của giảng viên chấm phản biện

.....

.....

.....

.....

2. Những mặt còn hạn chế

.....

.....

.....

.....

3. Ý kiến của giảng viên chấm phản biện

Được bảo vệ Không được bảo vệ Điểm phản biện

Hải Phòng, ngày.....tháng.....năm 2023

Giảng viên chấm phản biện
(ký và ghi rõ họ tên)

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN VỀ INTERNET VẠN VẬT	3
1.1. GIỚI THIỆU VỀ INTERNET VẠN VẬT	3
1.1.1. Giới thiệu về Internet of Things (IoT)	3
1.1.2. Lịch sử hình thành	3
1.2 CÔNG NGHỆ ETHERNET	7
1.2.1. Giới thiệu	7
1.2.2. Các đặc tính chung của Ethernet	8
1.2.3. Hoạt động của Ethernet	10
1.2.4. Các loại mạng Ethernet	12
a. Các hệ thống Ethernet 10Mb/s :	12
b. Các hệ thống Ethernet 100 Mb/s – Ethernet cao tốc (Fast Ethernet)	13
c. Các hệ thống Giga Ethernet	13
1.3. Arduino Mega 2560	13
1.4. Giới thiệu về ESP8266 NodeMCU	15
1.5. Arduino Ethernet Shield	17
Thông số kỹ thuật	18
CHƯƠNG 2. GIỚI THIỆU CÁC CẢM BIẾN	19
2.1. Biến đổi ADC	19
2.1.1. Nguyên lý hoạt động của ADC	19
a. ADC dạng flash (Flash ADC)	19
b. ADC xấp xỉ liên tiếp	20
c. ADC Sigma-Delta	20
2.1.2. Các thông số của ADC	21
a. Độ phân giải của ADC	21
b. Tốc độ lấy mẫu	21
c. Điện áp tham chiếu	21
2.2. MODULE CẢM BIẾN KHÍ GA MQ2	22
2.3. MODULE CẢM BIẾN KHÍ GA MQ135	23
CHƯƠNG 3. GIÁM SÁT NỒNG ĐỘ KHÍ GAS QUA INTERNET	
3.1. Thiết kế hệ thống	26
3.2. Cảm biến MQ135 và Arduino	27
3.3. Giám sát độ ẩm qua Internet	29
Kết luận	35
TÀI LIỆU THAM KHẢO	38

LỜI NÓI ĐẦU

Công nghệ Internet vạn vật IoT đang ngày càng được ứng dụng rộng rãi trong mọi mặt của cuộc sống. IoT cũng đang là động lực phát triển cho các lĩnh vực khác như điều khiển, giám sát thiết bị, hệ thống từ xa qua Internet, ví dụ như đo lường, cảnh báo và điều khiển hệ thống giám sát các thông số của môi trường. Với lý do đó em lựa chọn đề tài ‘Nghiên cứu, thiết kế hệ thống đo và giám sát nồng độ chất độc – hại trong không khí ứng dụng công nghệ Internet vạn vật ‘ làm đề án tốt nghiệp của mình.

Trong quá trình thực hiện đề án em đã được thầy Đoàn Hữu Chức tận tình hướng dẫn. Em chân thành cảm ơn những chỉ bảo hướng dẫn của thầy. Do kiến thức và thời gian làm đề án có hạn nên không tránh còn các thiếu sót em rất mong các thầy và các bạn góp ý để bản đề án được hoàn thiện hơn.

Em xin chân thành cảm ơn !

Hải Phòng, ngày tháng năm 2023

Sinh viên thực hiện

Kim Văn Đình

CHƯƠNG 1. TỔNG QUAN VỀ INTERNET VẠN VẬT

1.1. GIỚI THIỆU VỀ INTERNET VẠN VẬT

1.1.1. Giới thiệu về Internet of Things (IoT)

Khi nhu cầu phát triển các ứng dụng liên quan đến Internet ngày càng cao. Và IoT (Internet of things) là một công nghệ quan trọng mà tất cả các thiết bị có thể kết nối với nhau. Việc kết nối thì có thể thực hiện qua Wi-Fi, mạng viễn thông băng rộng (3G, 4G), Bluetooth, ZigBee, hồng ngoại... Về cơ bản, IoT là một hệ thống mạng lưới mà trong đó tất cả các thiết bị, đối tượng được kết nối Internet thông qua thiết bị mạng (network devices) hoặc các bộ định tuyến (routers). IoT cho phép các đối tượng được điều khiển từ xa dựa trên hệ thống mạng hiện tại. Công nghệ tiên tiến này giúp giảm công sức vận hành của con người bằng cách tự động hóa việc điều khiển các thiết bị.



Hình 1.1. Các thành phần chính trong một hệ thống IoT.

- Thiết bị: Mỗi thiết bị sẽ bao gồm một hoặc nhiều cảm biến để phát hiện các thông số của ứng dụng và gửi chúng đến Platform.
- IoT – Platform:
 - Nền tảng này là một phần mềm được lưu trữ trực tuyến còn được gọi là điện toán đám mây, các thiết bị được kết nối với nhau thông qua nó.
 - Nền tảng này thu thập dữ liệu từ thiết bị, toàn bộ dữ liệu được phân tích, xử lý, phát hiện nếu có lỗi phát sinh trong quá trình hệ thống vận hành.
- Kết nối Internet: Để giao tiếp được trong IoT, kết nối Internet của các thiết bị là một điều bắt buộc. Wifi là một trong những phương thức kết nối Internet phổ biến.
- Ứng dụng: Ứng dụng là giao diện để người dùng điều khiển.

1.1.2. Lịch sử hình thành

Khái niệm về một mạng lưới thiết bị được kết nối với nhau đã được thảo luận vào đầu năm 1982, với một máy bán hàng tự động Coke được thực hiện ở Đại học Carnegie Mellon trở thành thiết bị kết nối Internet đầu tiên trên thế giới. Thuật ngữ “Internet of things” được sử dụng lần đầu tiên bởi Kevin Ashton vào năm 1999. Sau đó IoT trải qua nhiều giai đoạn và có bước phát triển nhảy vọt cho đến ngày nay.



Hình 1.2. Lịch sử hình thành

➤ Nhà thông minh (Smart Home)

Bất cứ khi nào chúng ta nghĩ về các hệ thống IoT, ứng dụng quan trọng, hiệu quả và nổi bật nhất được nhắc đến chính là Smart Home – ứng dụng IOT xếp hạng cao nhất trên tất cả các kênh. Hiện nay do nhu cầu muốn được sở hữu căn hộ thông minh của người dùng ngày càng cao nên nhà thông minh là một trong những ứng dụng được nhiều người quan tâm .

Một ngôi nhà có thể giúp bạn quản lý các [thiết bị điện thông minh](#) điều khiển từ xa, thông qua internet hoặc các thiết bị điện tử bạn đang sử dụng như laptop, điện thoại,... Bạn sẽ có được sự nghỉ ngơi thoải mái với smarthome. Bạn không phải mất nhiều thời gian và công sức để đi lên đi xuống bật tắt điện, điều hòa, hay không phải đi ra đi vào để mở rèm cửa, mở cửa nhà, công... Tất cả có thể tự động thông qua hệ thống cảm ứng và hệ thống tự động. Bên cạnh đó, bạn còn có thể kiểm soát ngôi nhà của mình với hệ thống an ninh tự động, hệ thống giám sát từ xa,...



Hình 1.3. Nhà thông minh (Smart Home)

➤ **Giao thông thông minh**

An toàn là điều đầu tiên khi nghĩ đến tác động của IoT đối với giao thông vận tải. Ý tưởng đưa ra là các phương tiện có khả năng liên lạc với nhau bằng cách sử dụng dữ liệu đã được phân tích để có thể giảm đáng kể các sự cố tai nạn xảy ra khi tham gia giao thông. Sử dụng cảm biến, các phương tiện như ô tô, xe buýt được cảnh báo nguy cơ tiềm ẩn trên đường, hoặc thậm chí là tình trạng ùn tắc giao thông ở một số tuyến đường.

Dịch vụ vận chuyển hàng hóa cũng được ứng dụng từ công nghệ này. Công nghệ quản lý lịch trình vận chuyển, tối ưu hóa các tuyến giao hàng, mức tiêu thụ nhiên liệu của phương tiện, giám sát tốc độ của tài xế giao hàng tuân thủ quy định an toàn nhằm mang lại những lợi ích về kinh tế và sự hài lòng của khách hàng.

➤ **Y tế thông minh**

IoT có các ứng dụng khác nhau trong chăm sóc sức khỏe, từ các thiết bị giám sát từ xa đến các bộ cảm ứng tiên tiến và thông minh để tích hợp thiết bị. Nó có tiềm năng để cải thiện cách thức các bác sĩ chăm sóc và giữ cho bệnh nhân an toàn và khỏe mạnh.

Miếng dán theo dõi sức khỏe cho bệnh nhân: bạn không cần đến bác sĩ, những thông số về nhịp tim, huyết áp, đều được thu thập từ xa được phân tích sau đó chuẩn đoán để đưa ra tình trạng sức khỏe hiện tại của bệnh nhân và có thể dự đoán nguy cơ mắc bệnh nhằm có biện pháp phòng ngừa kịp thời.



Hình 1.4. Mô hình chăm sóc sức khỏe

➤ **Nông nghiệp (Smart Farming)**

Mô hình nhà kính là một trong những ứng dụng điển hình của công nghệ IoT được áp dụng trong lĩnh vực nông nghiệp. Và ở nước ta đã được áp dụng rộng rãi. Bên trong hệ thống này cây trồng hoàn toàn cách ly với điều kiện thời tiết bên ngoài, việc điều khiển nhiệt độ, độ ẩm, ánh sáng đều tự động hóa. Đồng thời theo dõi được tình trạng phát triển của cây trồng, xác định thời gian thu hoạch, giảm thiểu tối đa công suất người lao động.



Hình 1.5. Nông nghiệp (Smart Farming).

➤ **Thành phố thông minh (Smart City)**

Có thể xem đây là tập hợp của tất cả ứng dụng của IoT vào một hệ thống lớn. Một giải pháp đã và đang được nhiều quốc gia trên thế giới áp dụng ở các thành phố

lớn nhằm giải quyết những vấn đề cấp bách như tình trạng kẹt xe, gia tăng dân số, ô nhiễm môi trường, ngập lụt, ...

Mọi thứ trong thành phố thông minh này được kết nối, dữ liệu sẽ được giám sát bởi một loạt các máy tính mà không cần bất kỳ sự tương tác nào của con người.



Hình 1.6. Mô hình thành phố thông minh.

1.2 CÔNG NGHỆ ETHERNET

1.2.1. Giới thiệu

Ngày nay, Ethernet đã trở thành công nghệ mạng cục bộ được sử dụng rộng rãi. Sau 30 năm ra đời, công nghệ Ethernet vẫn đang được tiếp tục phát triển những khả năng mới đáp ứng những nhu cầu mới và trở thành công nghệ mạng phổ biến và tiện dụng.

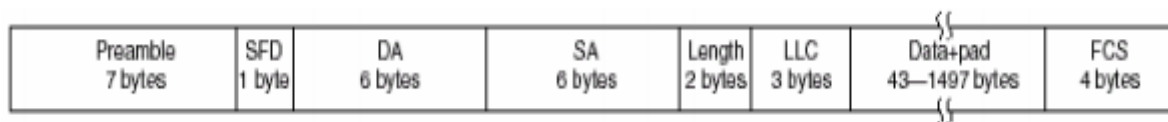
Ngày 22 tháng 5 năm 1973, Robert Metcalfe thuộc Trung tâm Nghiên cứu Palo Alto của hãng Xerox – PARC, bang California, đã đưa ra ý tưởng hệ thống kết nối mạng máy tính cho phép các máy tính có thể truyền dữ liệu với nhau và với máy in laser. Lúc này, các hệ thống tính toán lớn đều được thiết kế dựa trên các máy tính trung tâm đắt tiền (mainframe). Điểm khác biệt lớn mà Ethernet mang lại là các máy tính có thể trao đổi thông tin trực tiếp với nhau mà không cần qua máy tính trung tâm. Mô hình mới này làm thay đổi thế giới công nghệ truyền thông. Chuẩn Ethernet 10Mbps đầu tiên được xuất bản năm 1980 bởi sự phối hợp phát triển của 3 hãng DEC, Intel và Xerox. Chuẩn này có tên DIX Ethernet (lấy tên theo 3 chữ cái đầu của

tên các hãng). Ủy ban 802.3 của IEEE đã lấy DIX Ethernet làm nền tảng để phát triển. Năm 1985, chuẩn 802.3 đầu tiên đã ra đời với tên **IEEE 802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method versus Physical Layer Specification**. Mặc dù không sử dụng tên Ethernet nhưng hầu hết mọi người đều hiểu đó là chuẩn của công nghệ Ethernet. Ngày nay chuẩn IEEE 802.3 là chuẩn chính thức của Ethernet. IEEE đã phát triển chuẩn Ethernet trên nhiều công nghệ truyền dẫn khác nhau vì thế có nhiều loại mạng Ethernet.

1.2.2. Các đặc tính chung của Ethernet

a. Cấu trúc khung tin Ethernet

Các chuẩn Ethernet đều hoạt động ở tầng Data Link trong mô hình 7 lớp OSI vì thế đơn vị dữ liệu mà các trạm trao đổi với nhau là các khung (frame). Cấu trúc khung Ethernet như sau:



Hình 1.7. Cấu trúc khung tin Ethernet.

Các trường quan trọng trong phần mào đầu sẽ được mô tả dưới đây:

Preamble: trường này đánh dấu sự xuất hiện của khung bit, nó luôn mang giá trị 10101010. Từ nhóm bit này, phía nhận có thể tạo ra xung đồng hồ 10Mhz.

SFD (start frame delimiter): trường này mới thực sự xác định sự bắt đầu của 1 khung. Nó luôn mang giá trị 10101011.

Các trường Destination và Source: mang địa chỉ vật lý của các trạm nhận và gửi khung, xác định khung được gửi từ đâu và sẽ được gửi tới đâu.

LEN: giá trị của trường nói lên độ lớn của phần dữ liệu mà khung mang theo.

FCS mang CRC (cyclic redundancy checksum): phía gửi sẽ tính toán trường này trước khi truyền khung. Phía nhận tính toán lại CRC này theo cách tương tự. Nếu hai kết quả trùng nhau, khung được xem là nhận đúng, ngược lại khung coi như là lỗi và bị loại bỏ.

b. Cấu trúc địa chỉ Ethernet

Mỗi giao tiếp mạng Ethernet được định danh duy nhất bởi 48 bit địa chỉ (6 octet). Đây là địa chỉ được ấn định khi sản xuất thiết bị, gọi là địa chỉ MAC (Media Access Control Address).

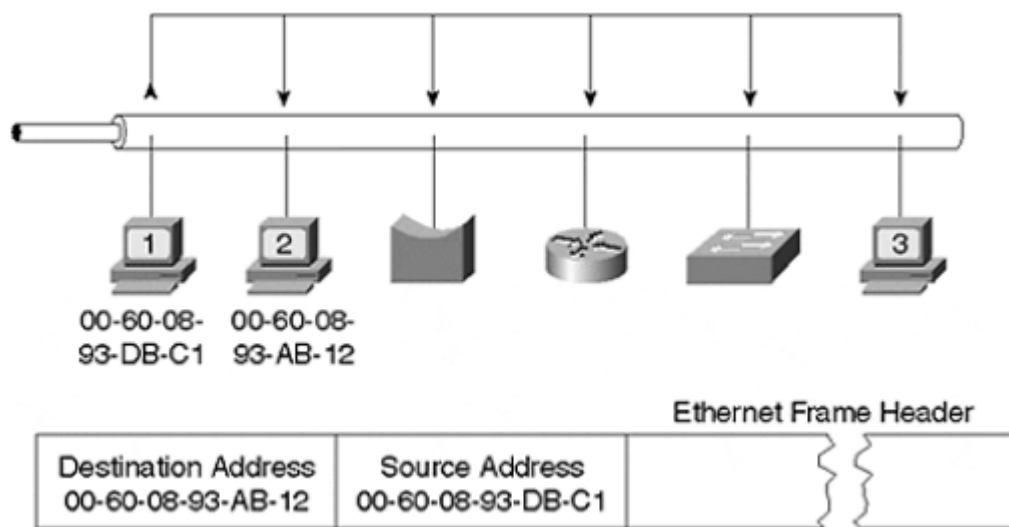
Địa chỉ MAC được biểu diễn bởi các chữ số hexa (hệ cơ số 16). Ví dụ : 00:60:97:8F:4F:86 hoặc 00-60-97-8F-4F-86. Khuôn dạng địa chỉ MAC được chia làm 2 phần:

- 3 octet đầu xác định hãng sản xuất, chịu sự quản lý của tổ chức IEEE.
- 3 octet sau do nhà sản xuất ấn định.

Kết hợp ta sẽ có một địa chỉ MAC duy nhất cho một giao tiếp mạng Ethernet. Địa chỉ MAC được sử dụng làm địa chỉ nguồn và địa chỉ đích trong khung Ethernet.

c. Các loại khung Ethernet

Các khung unicast. Giả sử trạm 1 cần truyền khung tới trạm 2



Hình 1.8. Các khung unicast.

Khung Ethernet do trạm 1 tạo ra có địa chỉ:

MAC nguồn: 00-60-08-93-DB-C1

MAC đích: 00-60-08-93-AB-12

Đây là khung unicast. Khung này được truyền tới một trạm xác định.

+ Tất cả các trạm trong phân đoạn mạng trên sẽ đều nhận được khung này nhưng:

+ Chỉ có trạm 2 thấy địa chỉ MAC đích của khung trùng với địa chỉ MAC của giao tiếp mạng của mình nên tiếp tục xử lý các thông tin khác trong khung.

+ Các trạm khác sau khi so sánh địa chỉ sẽ bỏ qua không tiếp tục xử lý khung nữa.

Các khung broadcast

Các khung broadcast có địa chỉ MAC đích là FF-FF-FF-FF-FF-FF (48 bit 1).

Khi nhận được các khung này, mặc dù không trùng với địa chỉ MAC của giao tiếp

mạng của mình nhưng các trạm đều phải nhận khung và tiếp tục xử lý. Giao thức ARP sử dụng các khung broadcast này để tìm địa chỉ MAC tương ứng với một địa chỉ IP cho trước.

Một số giao thức định tuyến cũng sử dụng các khung broadcast để các router trao đổi bảng định tuyến.

Các khung multicast

Trạm nguồn gửi khung tới một số trạm nhất định chứ không phải là tất cả. Địa chỉ MAC đích của khung là địa chỉ đặc biệt mà chỉ các trạm trong cùng nhóm mới chấp nhận các khung gửi tới địa chỉ này. Note: Địa chỉ MAC nguồn của khung luôn là địa chỉ MAC của giao tiếp mạng tạo ra khung. Trong khi đó địa chỉ MAC đích của khung thì phụ thuộc vào một trong ba loại khung nêu trên.

1.2.3. Hoạt động của Ethernet

Phương thức điều khiển truy nhập CSMA/CD quy định hoạt động của hệ thống Ethernet. Một số khái niệm cơ bản liên quan đến quá trình truyền khung Ethernet:

Khi tín hiệu đang được truyền trên kênh truyền, kênh truyền lúc này bận và ta gọi trạng thái này là có sóng mang – carrier.

Khi đường truyền rỗi: không có sóng mang – absence carrier.

Nếu hai trạm cùng truyền khung đồng thời thì chúng sẽ phát hiện ra sự xung đột và phải thực hiện lại quá trình truyền khung.

Khoảng thời gian để một giao tiếp mạng khôi phục lại sau mỗi lần nhận khung được gọi là khoảng trống liên khung (interframe gap) – ký hiệu IFG. Giá trị của IFG bằng 96 lần thời gian của một bit. Ethernet 10Mb/s: IFG = 9,6 us
Ethernet 100Mb/s: IFG = 960 ns Ethernet 1000Mb/s: IFG = 96 ns

Cách thức truyền khung và phát hiện xung đột diễn ra như sau:

1. Khi phát hiện đường truyền rỗi, máy trạm sẽ đợi thêm một khoảng thời gian bằng IFG, sau đó nó thực hiện ngay việc truyền khung. Nếu truyền nhiều khung thì giữa các khung phải cách nhau khoảng IFG.

2. Trong trường hợp đường truyền bận, máy trạm sẽ tiếp tục lắng nghe đường truyền cho đến khi đường truyền rỗi thì thực hiện lại 1.

3. Trường hợp khi quá trình truyền khung đang diễn ra thì máy trạm phát hiện thấy sự xung đột, máy trạm sẽ phải tiếp tục truyền 32 bit dữ liệu. Nếu sự xung đột

được phát hiện ngay khi mới bắt đầu truyền khung thì máy trạm sẽ phải truyền hết trường preamble và thêm 32 bit nữa, việc truyền nốt các bit này (ta xem như là các bit báo hiệu tắc nghẽn) đảm bảo tín hiệu sẽ tồn tại trên đường truyền đủ lâu cho phép các trạm khác (trong các trạm gây ra xung đột) nhận ra được sự xung đột và xử lý:

- Sau khi truyền hết các bit báo hiệu tắc nghẽn, máy trạm sẽ đợi trong một khoảng thời gian ngẫu nhiên hy vọng sau đó sẽ không gặp xung đột và thực hiện lại việc truyền khung như bước 1.

- Trong lần truyền khung tiếp theo này mà vẫn gặp xung đột, máy trạm buộc phải đợi thêm lần nữa với khoảng thời gian ngẫu nhiên nhưng dài hơn.

4. Khi một trạm truyền thành công 512 bit (không tính trường preamble), ta xem như kênh truyền đã bị chiếm. Điều này cũng có nghĩa là không thể có xung đột xảy ra nữa. Khoảng thời gian ứng với thời gian của 512 bit được gọi là slotTime. Đây là tham số quan trọng quyết định nhiều tới việc thiết kế. Do bản chất cùng chia sẻ kênh truyền, tại một thời điểm chỉ có một trạm được phép truyền khung. Càng có nhiều trạm trong phân đoạn mạng thì sự xung đột càng xảy ra nhiều, khi đó tốc độ truyền bị giảm xuống.

Sự xung đột là hiện tượng xảy ra bình thường trong hoạt động của mạng Ethernet (từ xung đột dễ gây hiểu nhầm là mạng bị sự cố hay là hoạt động sai, hỏng hóc).

Khái niệm slot Time

Trong ví dụ này, trạm 1 và trạm 2 được xem như hai trạm ở hai phía xa nhất của mạng. Trạm 1 truyền khung tới trạm 2, ngay trước khi khung này tới trạm 2, trạm 2 cũng quyết định truyền khung (vì nó thấy đường truyền rỗi).

Để mạng Ethernet hoạt động đúng, mỗi máy trạm phải phát hiện và thông báo sự xung đột tới trạm xa nhất trong mạng trước khi một trạm nguồn hoàn thành việc truyền khung.

Khung Ethernet kích cỡ nhỏ nhất là 512 bit (64 octet), do đó khoảng thời gian nhỏ nhất để phát hiện và thông báo xung đột là 512 lần thời gian một bit.

Ethernet 10Mb/s : slot Time = 51,2 us

Ethernet 100Mb/s : slot Time = 5,12 us

Ethernet 1000Mb/s : slot Time = 512 ns

Trường hợp vi phạm thời gian slotTime, mạng Ethernet sẽ hoạt động không đúng nữa. Mỗi lần truyền khung, máy trạm sẽ lưu khung cần truyền trong bộ đệm cho đến khi nó truyền thành công. Giả sử mạng không đáp ứng đúng tham số slotTime. Trạm 1 truyền 512 bit thành công không hề bị xung đột, lúc này khung được xem là truyền thành công và bị xoá khỏi bộ đệm. Do sự phát hiện xung đột bị trễ, trạm 1 lúc này muốn truyền lại khung cũng không được nữa vì khung đã bị xoá khỏi bộ đệm rồi. Mạng sẽ không hoạt động đúng.

Một mạng Ethernet được thiết kế đúng phải thoả mãn điều kiện sau: “ Thời gian trễ tổng cộng lớn nhất để truyền khung Ethernet từ trạm này tới trạm khác trên mạng phải nhỏ hơn một nửa slotTime”.

Thời gian trễ tổng cộng nói tới ở đây bao gồm trễ qua các thành phần truyền khung: trễ truyền tín hiệu trên cáp nối, trễ qua bộ repeater. Thời gian trễ của từng thành phần phụ thuộc vào đặc tính riêng của chúng. Các nhà sản xuất thiết bị ghi rõ và khi thiết kế cần lựa chọn và tính toán để thoả mãn điều kiện hoạt động đúng của mạng Ethernet.



Hình 1.9. Hai trạm phía xa nhất mạng Ethernet.

1.2.4. Các loại mạng Ethernet

IEEE đã phát triển chuẩn Ethernet trên nhiều công nghệ truyền dẫn khác nhau vì thế có nhiều loại mạng Ethernet. Mỗi loại mạng được mô tả dựa theo ba yếu tố: tốc độ, phương thức tín hiệu sử dụng và đặc tính đường truyền vật lý.

a. Các hệ thống Ethernet 10Mb/s :

10Base5. Đây là tiêu chuẩn Ethernet đầu tiên, dựa trên cáp đồng trục loại dày. Tốc độ đạt được 10 Mb/s, sử dụng băng tần cơ sở, chiều dài cáp tối đa cho 1 phân đoạn mạng là 500m.

10Base2. Có tên khác là “thin Ethernet” , dựa trên hệ thống cáp đồng trục mỏng với tốc độ 10 Mb/s, chiều dài cáp tối đa của phân đoạn là 185 m (IEEE làm tròn thành 200m).

10BaseT. Chữ T là viết tắt của “twisted”: cáp xoắn cặp. 10BaseT hoạt động tốc độ 10 Mb/s dựa trên hệ thống cáp xoắn cặp Cat 3 trở lên.

10BaseF. F là viết tắt của Fiber Optic (sợi quang). Đây là chuẩn Ethernet dùng cho sợi quang hoạt động ở tốc độ 10 Mb/s , ra đời năm 1993.

b. Các hệ thống Ethernet 100 Mb/s – Ethernet cao tốc (Fast Ethernet)

100BaseT. Chuẩn Ethernet hoạt động với tốc độ 100 Mb/s trên cả cáp xoắn cặp lẫn cáp sợi quang.

100BaseX. Chữ X nói lên đặc tính mã hóa đường truyền của hệ thống này (sử dụng phương pháp mã hoá 4B/5B của chuẩn FDDI). Bao gồm 2 chuẩn 100BaseFX và 100BaseTX

- 100BaseFX. Tốc độ 100Mb/s, sử dụng cáp sợi quang đa mode.

- 100BaseTX. Tốc độ 100Mb/s, sử dụng cáp xoắn cặp.

100BaseT2 và 100BaseT4. Các chuẩn này sử dụng 2 cặp và 4 cặp cáp xoắn cặp Cat 3 trở lên tuy nhiên hiện nay hai chuẩn này ít được sử dụng.

c. Các hệ thống Giga Ethernet

1000BaseX. Chữ X nói lên đặc tính mã hoá đường truyền (chuẩn này dựa trên kiểu mã hoá 8B/10B dùng trong hệ thống kết nối tốc độ cao Fibre Channel được phát triển bởi ANSI). Chuẩn 1000BaseX gồm 3 loại:

- 1000Base-SX: tốc độ 1000 Mb/s, sử dụng sợi quang với sóng ngắn.

- 1000Base-LX: tốc độ 1000 Mb/s, sử dụng sợi quang với sóng dài.

- 1000Base-CX: tốc độ 1000 Mb/s, sử dụng cáp đồng.

1000BaseT. Hoạt động ở tốc độ Giga bit, băng tần cơ sở trên cáp xoắn cặp Cat 5 trở lên. Sử dụng kiểu mã hoá đường truyền riêng để đạt được tốc độ cao trên loại cáp này.

1.3. Arduino Mega 2560

Arduino Mega2560 là một hệ thống sử dụng vi điều khiển ATmega2560.

Hệ thống bao gồm:

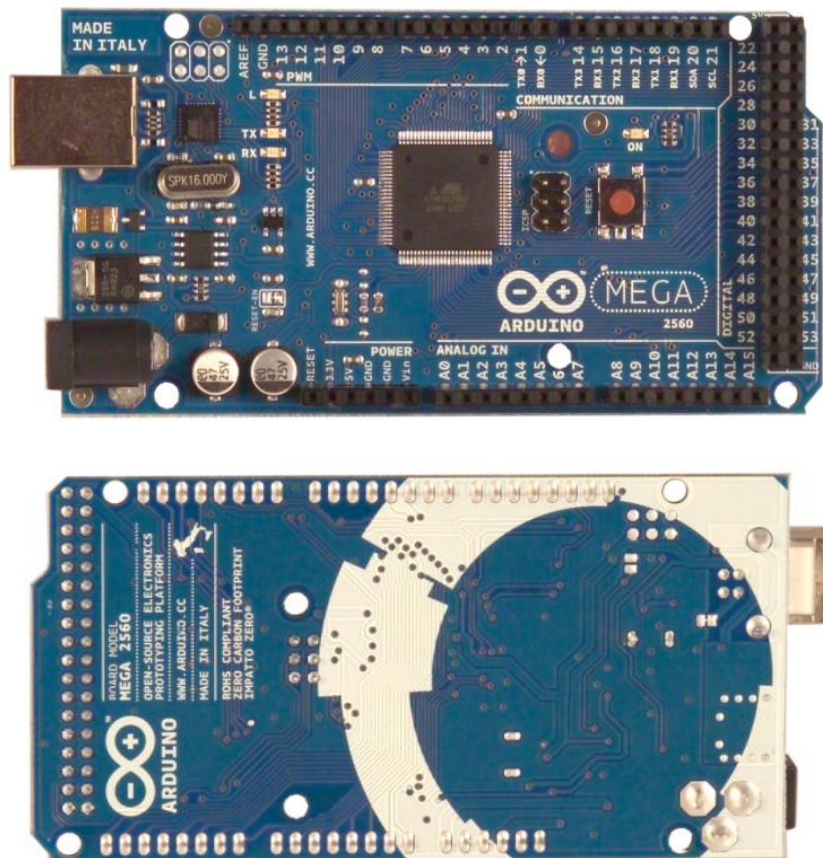
- 54 chân digital (15 có thể được sử dụng như các chân PWM)
- 16 đầu vào analog,

- UARTs (cổng nối tiếp phần cứng),
- thạch anh 16 MHz,
- cổng kết nối USB,
- jack cắm điện,
- 1 đầu ICSP,
- 1 nút reset.

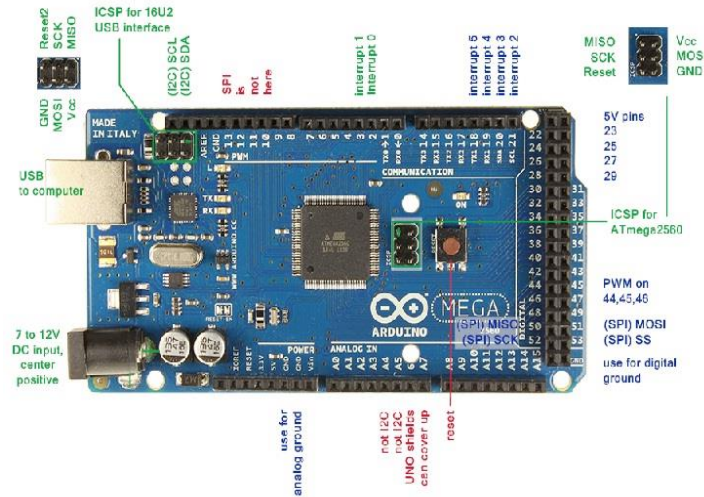
Nó chứa tất cả mọi thứ cần thiết để hỗ trợ các vi điều khiển.

Arduino Mega2560 khác với tất cả các vi xử lý trước giờ vì không sử dụng FTDI chip điều khiển chuyên tín hiệu từ USB để xử lý. Thay vào đó, nó sử dụng ATmega16U2 lập trình như là một công cụ chuyên đổi tín hiệu từ USB. Như vậy các kit này sẽ được lập trình và nạp trực tiếp qua các cổng USB. Ngoài ra, Arduino Mega2560 cơ bản vẫn giống Arduino Uno R3, chỉ khác số lượng chân và có nhiều tính năng mạnh mẽ hơn, nên vẫn có thể lập trình cho vi điều khiển này bằng chương trình lập trình cho Arduino Uno R3[2,3,8].

Sơ đồ chi tiết chân vào/ra của hệ thống Arduino Mega 2560 như hình 1.10 và 1.11 dưới đây.



Hình 1.10. Mặt trước và sau của Arduino Mega 2560 thực tế.



Hình 1.11. Bố trí chân vào/ra Arduino Mega 2560 thực tế.

Bảng 1.1. Thông số kỹ thuật của Arduino Mega 2560

Vi điều khiển	ATmega2560
Điện áp hoạt động	5V
Điện áp vào (đề nghị)	7V-15V
Điện áp vào (giới hạn)	6V-20V
Cường độ dòng điện trên mỗi 3.3V pin	50 mA
Cường độ dòng điện trên mỗi I/O pin	20 mA
Flash Memory	256 KB
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 Hz

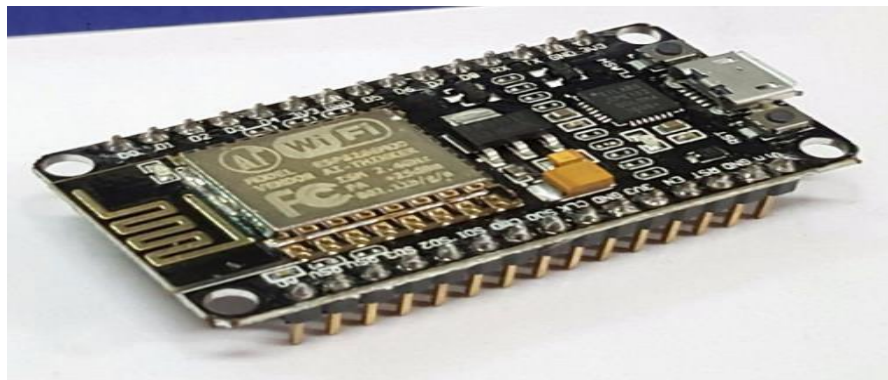
1.4. Giới thiệu về ESP8266 NodeMCU

Kít ESP8266 là kít phát triển dựa trên nền chip Wifi SoC ESP8266 với thiết kế dễ dàng sửa dụng vì tích hợp sẵn mạch nạp sử dụng chip CP2102 trên borad. Bên trong ESP8266 có sẵn một lõi vi xử lý vì thế bạn có thể trực tiếp lập trình cho ESP8266 mà không cần thêm bất kì con vi xử lý nào nữa.

Hiện tại có hai ngôn ngữ có thể lập trình cho ESP8266, sử dụng trực tiếp phần mềm IDE của Arduino để lập trình với bộ thư viện riêng hoặc sử dụng phần mềm node MCU và là dòng chip tích hợp Wi-Fi 2.4Ghz có thể lập trình được, rẻ tiền được sản xuất bởi một công ty bán dẫn Trung Quốc: Espressif Systems. Được phát hành đầu tiên vào tháng 8 năm 2014, đóng gói đưa ra thị trường dạng Module ESP-01.

Có khả năng kết nối Internet qua mạng Wi-Fi một cách nhanh chóng và sử dụng rất ít linh kiện đi kèm. Với giá cả có thể nói là rất rẻ so với tính năng và khả năng ESP8266 có thể làm được. ESP8266 có một cộng đồng các nhà phát triển trên thế giới rất lớn, cung cấp nhiều Module lập trình mã mở giúp nhiều người có thể tiếp cận và xây dựng ứng dụng rất nhanh.

Hiện nay tất cả các dòng chip ESP8266 trên thị trường đều mang nhãn ESP8266EX, là phiên bản nâng cấp của ESP8266, đã có hơn 14 phiên bản ESP ra đời, trong đó phổ biến nhất là ESP-12.



Hình 1.12. Hình ảnh thực tế của Chip NODEMCU ESP8266

Cấu tạo của NODEMCU ESP8266

Module ESP8266 có các chân dùng để cấp nguồn và thực hiện kết nối.

Chức năng của các chân như sau: + VCC: 3.3V lên đến 300Ma + GND: Chân Nối đất .

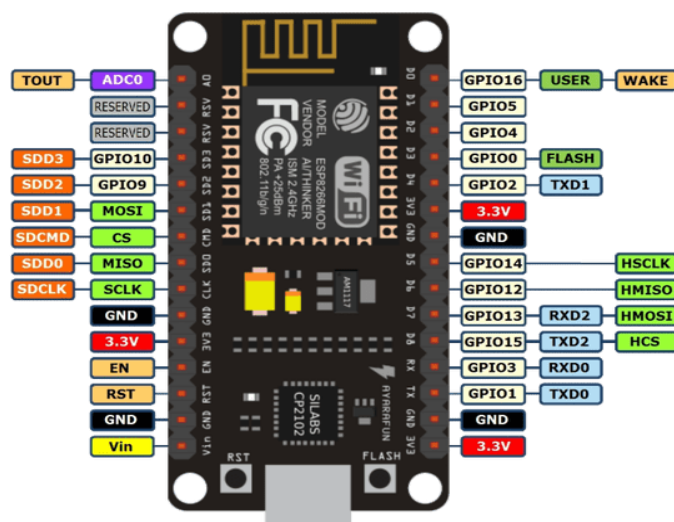
+ Tx: Chân Tx của giao thức UART, kết nối đến chân Rx của vi điều khiển.

+ Rx: Chân Rx của giao thức UART, kết nối đến chân Tx của vi điều khiển. + RST: chân reset, kéo xuống mass để reset.

+ 10 chân GPIO từ D0 – D8, có chức năng PWM, IIC, giao tiếp SPI, 1-Wire và ADC trên chân A0

+ Kết nối mạng wifi (có thể là sử dụng như điểm truy cập và/hoặc trạm máy chủ lưu trữ một, máy chủ web), kết nối internet để lấy hoặc tải lên dữ liệu.

Module ESP-12 kết hợp với firmware ESP8266 trên Arduino và thiết kế phần cứng giao tiếp tiêu chuẩn đã tạo nên NodeMCU, loại Kit phát triển ESP8266 phổ biến nhất trong thời điểm hiện tại. Với cách sử dụng, kết nối dễ dàng, có thể lập trình, nạp chương trình trực tiếp trên phần mềm Arduino, đồng thời tương thích với các bộ thư viện Arduino sẵn có.



Hình 1.13. Hình ảnh sơ đồ chân kết nối ESP8266

Tính năng của NODEMCU ESP8266

➤ Thông số kỹ thuật:

- IC chính: ESP8266 Wifi SoC.
- Phiên bản firmware: NodeMCU Lua
- Chip nạp và giao tiếp UART: CP2102.
- GPIO tương thích hoàn toàn với firmware Node MCU.
- Cấp nguồn: 5VDC MicroUSB hoặc Vin.
- GIPO giao tiếp mức 3.3VDC
- Tích hợp Led báo trạng thái, nút Reset, Flash.
- Tương thích hoàn toàn với trình biên dịch Arduino.
- Kích thước: 25 x 50 mm

1.5. *Arduino Ethernet Shield*

Arduino Ethernet W5100 của hãng PNLAB sử dụng chip Wiznet W5100 ethernet. Module này cho phép một Board Arduino có thể kết nối Internet. Chip Wiznet W5100 Ethernet cung cấp một địa chỉ mạng (IP) tương thích với cả TCP và UDP. Nó hỗ trợ tối đa 4 thiết bị kết nối đồng thời. Ethernet W5100 có thể tự động nhận IP từ router bằng giao thức DHCP. Bạn không mất nhiều thời gian để thiết đặt IP tĩnh cho Ethernet Shield.

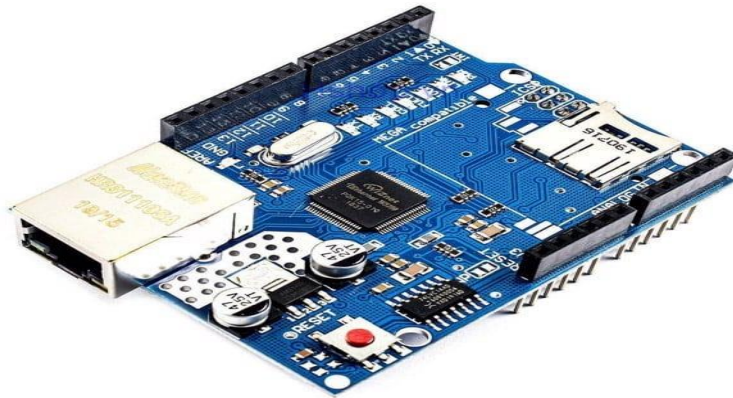
Ethernet Shield hoàn toàn tương thích với Arduino [Uno](#) và [Mega](#), giao tiếp với Arduino qua giao thức SPI, chân digital 10 trên arduino được sử dụng để xác định kết nối cho ethernet shield. Vì vậy, bạn không thể sử dụng port 10 khi cắm Ethernet Shield vào Arduino.

IC điều khiển W5100 trên Arduino Ethernet Shield có thể thực hiện truyền dữ liệu thông qua 2 giao thức là TCP và UDP. Số đường truyền dữ liệu song song tối đa là 4, khả năng truyền song song cùng lúc 4 luồng dữ liệu giúp board có khả năng nhận dữ liệu từ internet với tỉ lệ lỗi thấp hơn (nguyên nhân thường là do mất dữ liệu trên đường truyền hoặc do thời gian truyền vượt quá giới hạn - time out).

Ngoài ra trên Ethernet Shield còn có Module đọc/ghi dữ liệu vào thẻ MicroSD sử dụng chân digital 4 để xác định kết nối.

Thông số kỹ thuật

- Chip sử dụng: Wiznet W5100 Ethernet với bộ đệm 16K
- Hãng sản xuất: PNLAB
- Điện áp hoạt động: 5VDC (Nguồn cấp từ Arduino)
- Tốc độ kết nối: 10 - 100 Mb
- Giao tiếp: SPI
- Sử dụng các trình điều khiển Ethernet chuẩn Arduino
- Hỗ trợ thẻ TF lên đến 16GB
- Hỗ trợ jack Ethernet chuẩn RJ45 kết nối mạng
- Kích thước: 7cm x 5.4cm x 2.4cm



Hình 1.14. Hình ảnh Arduino Ethernet Shield

CHƯƠNG 2. GIỚI THIỆU CÁC CẢM BIẾN

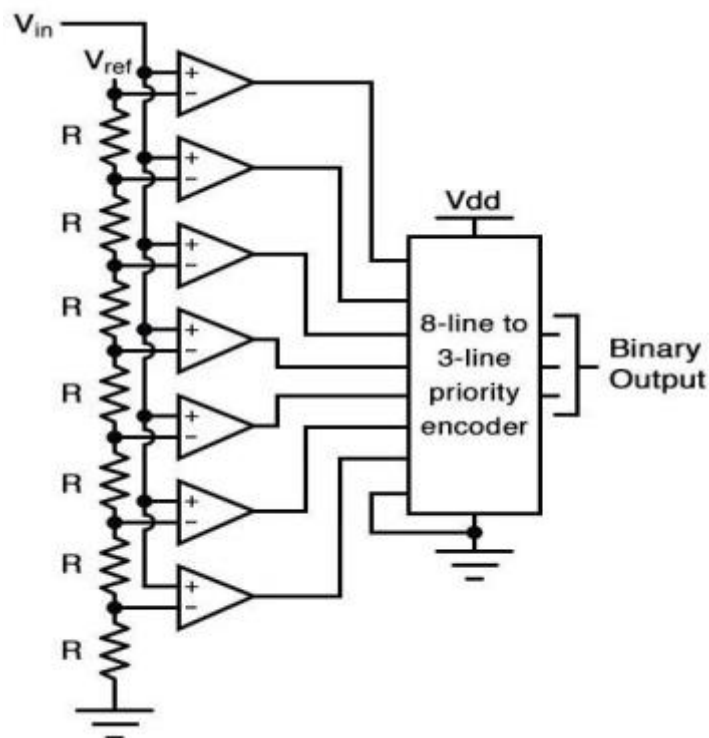
2.1. Biến đổi ADC

ADC là từ viết tắt của Analog to Digital Converter hay bộ chuyển đổi analog sang kỹ thuật số là một mạch chuyển đổi giá trị điện áp liên tục (analog) sang giá trị nhị phân (kỹ thuật số) mà thiết bị kỹ thuật số có thể hiểu được sau đó có thể được sử dụng để tính toán kỹ thuật số. Mạch ADC này có thể là vi mạch ADC hoặc được nhúng vào một bộ vi điều khiển.

2.1.1. Nguyên lý hoạt động của ADC

a. ADC dạng flash (Flash ADC)

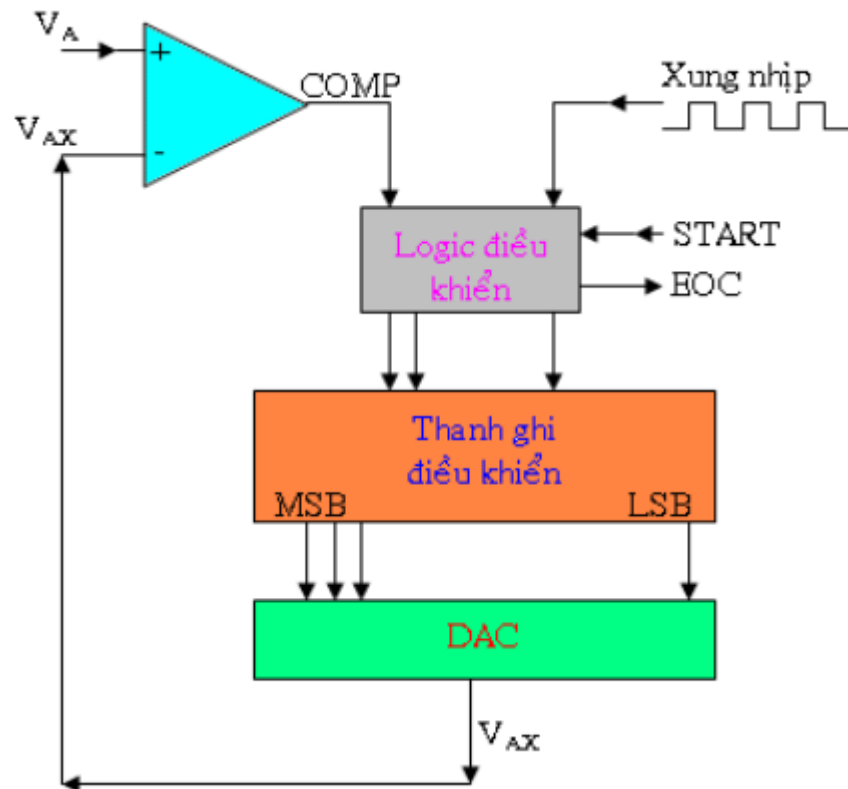
Flash ADC là dạng đơn giản nhất, thực hiện bằng dãy điện trở phân áp và các comparator điện áp. Nó là minh họa nhập đề cho hoạt động của ADC. Trong hình vẽ là ADC 16 mức "không âm", thực hiện bằng 15 comparator. Kết quả so được mạch lập mã Encoder tiếp nhận và chuyển sang mã nhị phân, trong trường hợp này là 4 bit. Nhịp lấy mẫu do phần nhận mã tự quyết định, và có thể đạt rất cao. Thay cho Bậc số hóa phải dùng mức số hoá (nếu số mức không trùng vào số 2M).



Hình 2.1. Cấu trúc ADC flash 8 bit.

b. ADC xấp xỉ liên tiếp

Bộ chuyển đổi liên tiếp - xấp xỉ (Successive Approximation Converter - SAC) là một trong những loại ADC thông dụng nhất. SAC có sơ đồ phức tạp hơn nhiều so với ADC dạng sóng bậc thang. Ngoài ra SAC còn có giá trị tc cố định, không phụ thuộc vào giá trị của đầu vào tương tự. Hình 2.2 là một cấu hình cơ bản của SAC, tương tự cấu hình của ADC dạng sóng bậc thang. Tuy nhiên ADC không sử dụng bộ đếm cung cấp đầu vào cho DAC mà thay vào đó là thanh ghi. Logic điều khiển sửa đổi nội dung lưu trên thanh ghi theo từng bit một cho đến khi dữ liệu ở thanh ghi biến thành giá trị số tương đương với đầu vào tương tự V_A trong phạm vi độ phân giải của bộ chuyển đổi.



Hình 2.2. Cấu trúc ADC xấp xỉ liên tiếp.

c. ADC Sigma-Delta

Bộ ADC này có lẽ là chính xác nhất. Nó bao gồm một bộ so sánh, một bộ DAC flash đơn giản và một thanh ghi bộ nhớ. Thiết bị ban đầu giả định tất cả các bit trong thanh ghi ngoại trừ bit có ý nghĩa cao nhất (là một) là số 0. Sau đó thanh ghi này sẽ gửi nó đến DAC chuyển nó thành điện áp analog, được so sánh với đầu vào thông qua bộ so sánh. Nếu điện áp đầu vào cao hơn điện áp DAC,

thì MSB vẫn là một. Quá trình này lặp lại cho đến khi tất cả các bit có được đặt thành không hoặc một, nói cách khác cho đến khi giá trị thanh ghi bằng điện áp đầu vào. ADC này là một trong những ADC thường được sử dụng nhất khi cần độ chính xác và tốc độ không quá giới hạn, ví dụ như trong vi điều khiển ADC loại SA có thể dễ dàng đạt được thời gian chuyển đổi vài micro giây.

2.1.2. Các thông số của ADC

a. Độ phân giải của ADC

Trong ADC dạng sóng bậc thang có nhiều yếu tố ảnh hưởng đến sai số của quá trình chuyển đổi như: kích cỡ bậc thang, tức độ phân giải của DAC cài trong đơn vị nhỏ nhất. Nếu giảm kích cỡ bậc thang ta có thể hạn chế bớt sai số nhưng luôn có khoảng cách chênh lệch giữa đại lượng thực tế và giá trị gán cho nó. Đây gọi là sai số lượng tử.

Cũng như trong ADC, độ chính xác không ảnh hưởng đến độ phân giải nhưng lại tùy thuộc vào độ chính xác của linh kiện trong mạch như: bộ so sánh, điện trở chính xác và chuyển mạch dòng của DAC, nguồn điện quy chiếu,... Mức sai số = 0.01% giá trị cực đại (đầy thang) cho biết kết quả ra từ ADC có thể sai biệt một khoảng như thế, do các linh kiện không lý tưởng.

b. Tốc độ lấy mẫu

Số lượng chuyển đổi từ analog sang kỹ thuật số mà bộ chuyển đổi có thể thực hiện mỗi giây được gọi là tốc độ mẫu. Ví dụ: một bộ ADC thực sự tốt có thể có tốc độ mẫu là 300M/ s. Đơn vị này được đọc là megasamples trên giây, nghĩa là một triệu mẫu mỗi giây. Lưu ý rằng tiền tố SI áp dụng ở đây. Tốc độ lấy mẫu phụ thuộc hoàn toàn vào loại bộ chuyển đổi và độ chính xác cần thiết. Nếu cần đọc rất chính xác, ADC thường dành nhiều thời gian hơn để xem xét tín hiệu đầu vào (thường là lấy mẫu và giữ hoặc đầu vào tích hợp) và nếu không cần độ chính xác cao thì nó có thể đọc rất nhanh.

c. Điện áp tham chiếu

Tất nhiên, không có ADC nào là tuyệt đối, vì vậy điện áp được ánh xạ tới giá trị nhị phân lớn nhất được gọi là điện áp tham chiếu. Ví dụ: trong bộ chuyển đổi 10 bit với 5V làm điện áp tham chiếu, 1111111111 (tất cả các bit một, số nhị

phân 10 bit cao nhất có thể) tương ứng với 5V và 0000000000 (số thấp nhất tương ứng với 0V). Vì vậy, mỗi bước nhị phân lên đại diện cho khoảng 4,9mV, vì có thể có 1024 chữ số trong 10 bit. Số đo điện áp trên mỗi bit này được gọi là độ phân giải của ADC. Điều gì sẽ xảy ra nếu điện áp thay đổi dưới 4,9mV mỗi bước? Nó sẽ đặt ADC vào vùng chết, do đó kết quả chuyển đổi luôn có một lỗi nhỏ. Có ngăn chặn lỗi này bằng cách sử dụng ADC có độ phân giải cao hơn ví dụ như bộ ADC lên đến 24 bit, mặc dù tần số chuyển đổi thấp.

2.2. MODULE CẢM BIẾN KHÍ GA MQ2

Module cảm biến khí ga MQ2 là một trong những loại cảm biến được sử dụng để nhận biết nhiều loại khí như: Butan, Propane, Methane, Alcohol, khói và khí gas. MQ2 là cảm biến có độ nhạy cao, thời gian đáp ứng nhanh. Bộ phận cảm nhận của cảm biến khí MQ2 là SnO₂, có độ dẫn điện thấp trong không khí sạch. Khi phát hiện các khí dễ cháy, độ dẫn điện của cảm biến sẽ tăng khi nồng độ khí tăng.

Module MQ2 bao gồm cảm iến khí gas MQ2, một chiết áp điều chỉnh độ nhạy, LM393, đèn LED báo hiệu. Dưới đây là hình ảnh của mô đun cảm biến khí gas MQ2:



Hình 2.3. Cảm biến khí gas MQ2

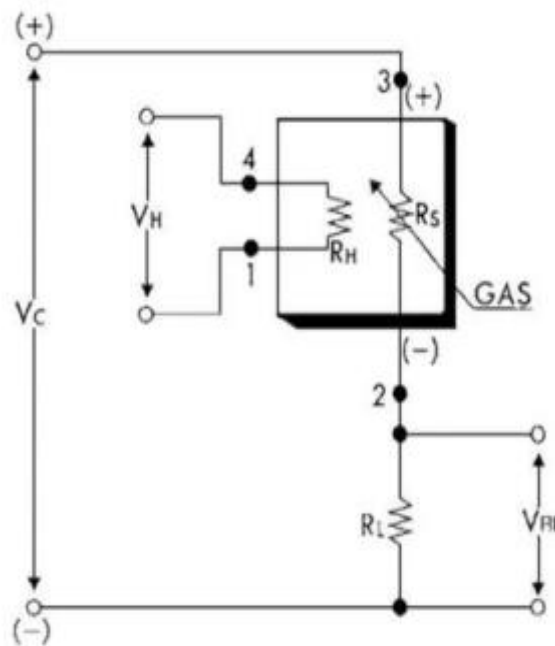
Thông số kỹ thuật :

- Nguồn cung cấp: 4.5V đến 5V DC

- Độ nhạy cao với khí Propane, khói và Butan
- Hỗ trợ đầu ra analog và digital
- Thời gian đáp ứng: $\leq 10s$
- Trở kháng khi nóng: $31\Omega \pm 3\Omega$
- Dòng tiêu thụ khi nóng: $\leq 180mA$
- Tích hợp LED ảo động trên modul
- Giao tiếp đơn giản với 4 chân VCC GND Analog và Digital. Người dùng có thể lựa chọn việc sử dụng chỉ một trong 2 chân analog và digital để lấy dữ liệu từ cảm biến.

2.3. MODULE CẢM BIẾN KHÍ GA MQ135

Cảm biến MQ135 thuộc dòng cảm biến bán dẫn, phần tử cảm biến là một Oxit có trở kháng có thể thay đổi với các khí khác nhau ở các nồng độ khác nhau. Thông thường cảm biến được mắc nối tiếp với một biến trở R_L để thay đổi độ nhạy như hình 2.4.



Hình 2.4. Sơ đồ nguyên lý cảm biến MQ 135.

Trong quá trình đo, cảm biến khí được đốt nóng bằng sợi đốt R_H với điện áp V_H , khi các khí cần đo tiếp xúc với cảm biến nó sẽ làm thay đổi giá trị điện trở của cảm biến R_S dẫn tới sự thay đổi của điện áp trên cảm biến. Giá trị điện áp V_{RL} được đọc bởi vi điều khiển và được dùng để tính toán ra nồng độ của khí.

Cảm biến có độ nhạy khác nhau với các loại khí khác nhau và được thể hiện trên đường cong đặc tính của từng loại cảm biến. Việc quy đổi giá trị điện áp từ cảm biến ra nồng độ khí cần đo được tính toán thông qua đồ thị đường cong đặc tính này với các giá trị R_0 và R_s tương ứng với mỗi cảm biến. Trên modul cảm biến MQ135, phần tử cảm biến được mắc nối tiếp với một trở tải R_L , giá trị của R_L được lựa chọn tối ưu cho từng cảm biến, với cảm biến MQ135 giá trị R_L được khuyến cáo là $10k\Omega$. Giá trị được đo về trên modul cảm biến chính là giá trị ADC của. Do R_L mắc nối tiếp với cảm biến nên:

$$V_s = V_{ref} - V_{RL}$$

Trong đó: V_s : Điện áp trên phần tử của cảm biến.

V_{ref} : Điện áp cấp cho cảm biến.

V_{RL} : Điện áp trên trở tải.

Theo định luật ôm ta có:

$$V_s/R_s = V_{RL}/R_L$$

Từ đó tính được:

$$R_s = \frac{R_L \times V_s}{V_{RL}} = \frac{R_L \times (V_{ref} - V_{RL})}{V_{RL}}$$

Giá trị R_0 được tính toán tương tự tại một nồng độ cho trước của khí cần đo (nồng độ để hiệu chuẩn cảm biến) trong môi trường không khí sạch, không có sự hiện diện của các loại khí khác. Nồng độ của khí được tính dựa theo:

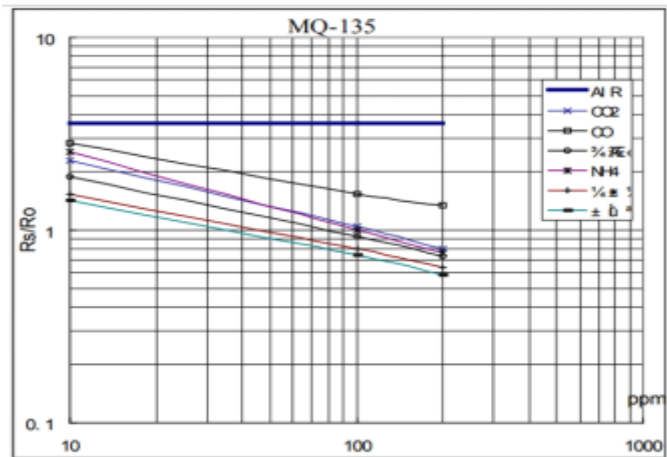
$$C = X_0 \times \left(\frac{R_s/R_0}{Y_0} \right)^m$$

Trong đó m là độ dốc của đường cong đặc trưng được tính toán theo công thức:

$$m = \frac{\log y_2 - \log y_1}{\log x_2 - \log x_1}$$

Đặc tính ra của MQ135 được cho trên hình 2.5 (đặc tính của nhà sản xuất cung cấp). Trong đó (x_1, y_1) , (x_2, y_2) là hai điểm bất kì được lựa chọn trên một phần của đường đặc trưng, nên cần lựa chọn các giá trị của (x_1, y_1) , (x_2, y_2) từ các phần tương ứng.

X_0 và Y_0 là nồng độ chất lượng khí và giá trị R_s/R_0 tại một điểm được chọn trên đường đặc trưng.



Hình 2.5. Đường đặc trưng của cảm biến MQ 135.

CHƯƠNG 3. GIÁM SÁT NỒNG ĐỘ KHÍ GAS QUA INTERNET

Trong chương này, em trình bày một số kết quả thực hiện hệ thống dùng cảm biến đo nồng độ khí Gas MQ135 hay còn gọi là cảm biến chất lượng không khí kết nối với Arduino sau đó truyền tải các thông số đo được qua hệ thống Internet. Hệ thống đo và giám sát nồng độ khí Gas được thiết kế dựa trên ý tưởng là sử dụng cảm biến để đo nồng độ khí độc hại trong môi trường. Nếu nồng độ khí độc hại vượt quá mức cho phép thì hệ thống sẽ tự động đưa ra giải pháp cải thiện chất lượng môi trường không khí, ví dụ như trong đồ án lựa chọn là khởi động hệ thống quạt thông gió để làm thay đổi không khí của môi trường. Hệ thống cảm biến đo nồng độ khí Gas MQ135 và điều khiển thiết bị ứng dụng IoT được trình bày một cách chi tiết.

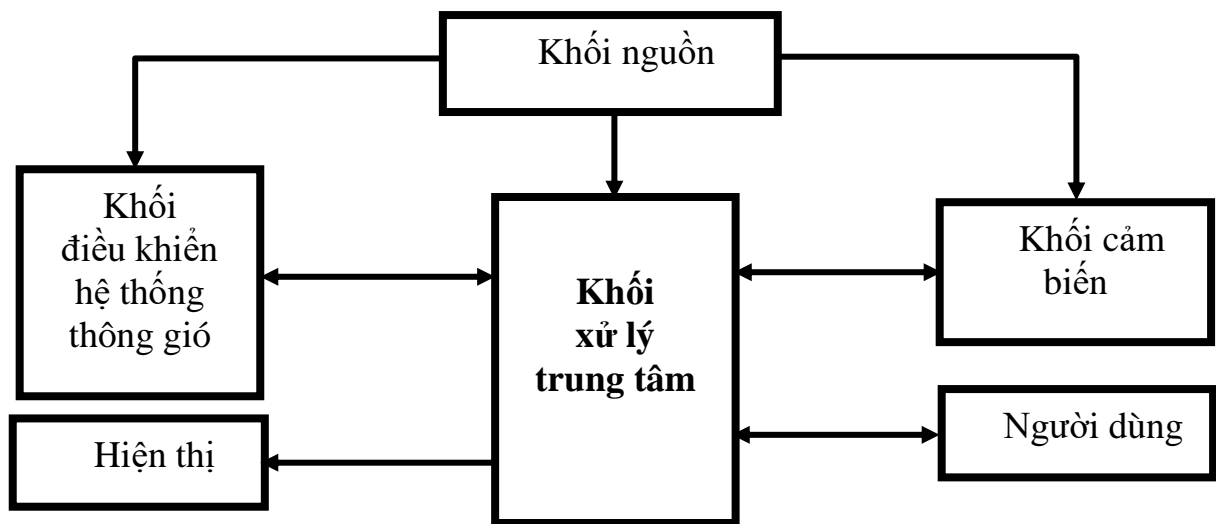
Hệ thống hoạt động dựa trên sự kết hợp của Module Arduino Ethernet Shield KIT Arduino 2560 và ứng dụng Blink IoT trên smartphone. Ứng dụng Blink IoT ngoài chức năng lưu trữ dữ liệu từ mạch đo gửi lên còn có chức năng hiển thị giao diện điều khiển thiết bị, dữ liệu về nồng độ khí Gas ra giao diện người dùng.

Khi nhận được tín hiệu từ cảm biến báo về thì bộ vi xử lý của hệ thống sẽ xử lý tín hiệu và sau đó truyền lên sever blynk thông qua giao diện Ethernet.

Sử dụng hệ thống gồm Module Arduino Ethernet Shield, KIT Arduino 2560 gửi tín hiệu về app android ta có thể truy cập vào hệ thống để điều khiển thiết bị trong môi trường không khí cần giám sát từ xa có mạng internet.

3.1. Thiết kế hệ thống

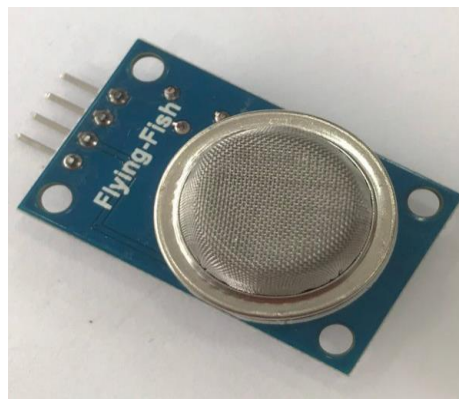
Hệ thống đo nồng độ khí độc hại và điều khiển thiết bị quạt thông gió về cơ bản gồm ba thành phần chính là: Khối cảm biến MQ135, Khối chấp hành cho điều khiển thiết bị thông gió và khối điều khiển trung tâm. Ngoài ra còn có thêm các khối như nguồn nuôi, hiển thị,...



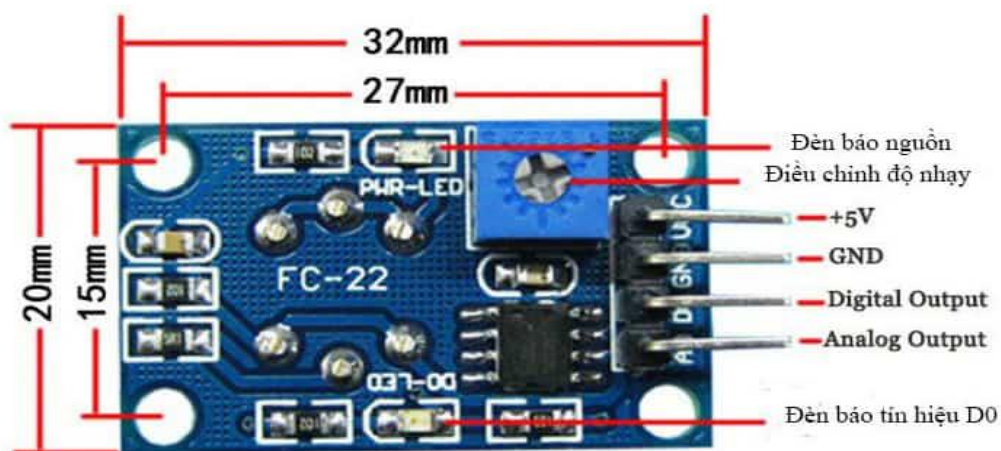
Hình 3.1. Sơ đồ khối hệ thống đo nồng độ khí độc hại

3.2. Cảm biến MQ135 và Arduino

Cảm biến này thường được dùng trong các thiết bị kiểm tra chất lượng không khí bên trong cao ốc, văn phòng, thích hợp để phát hiện NH₃, NO_x, Ancol, Benzen, khói, CO₂,...



Hình 3.2. Hình ảnh thực tế cảm biến MQ 135 đo nồng độ khí độc hại.



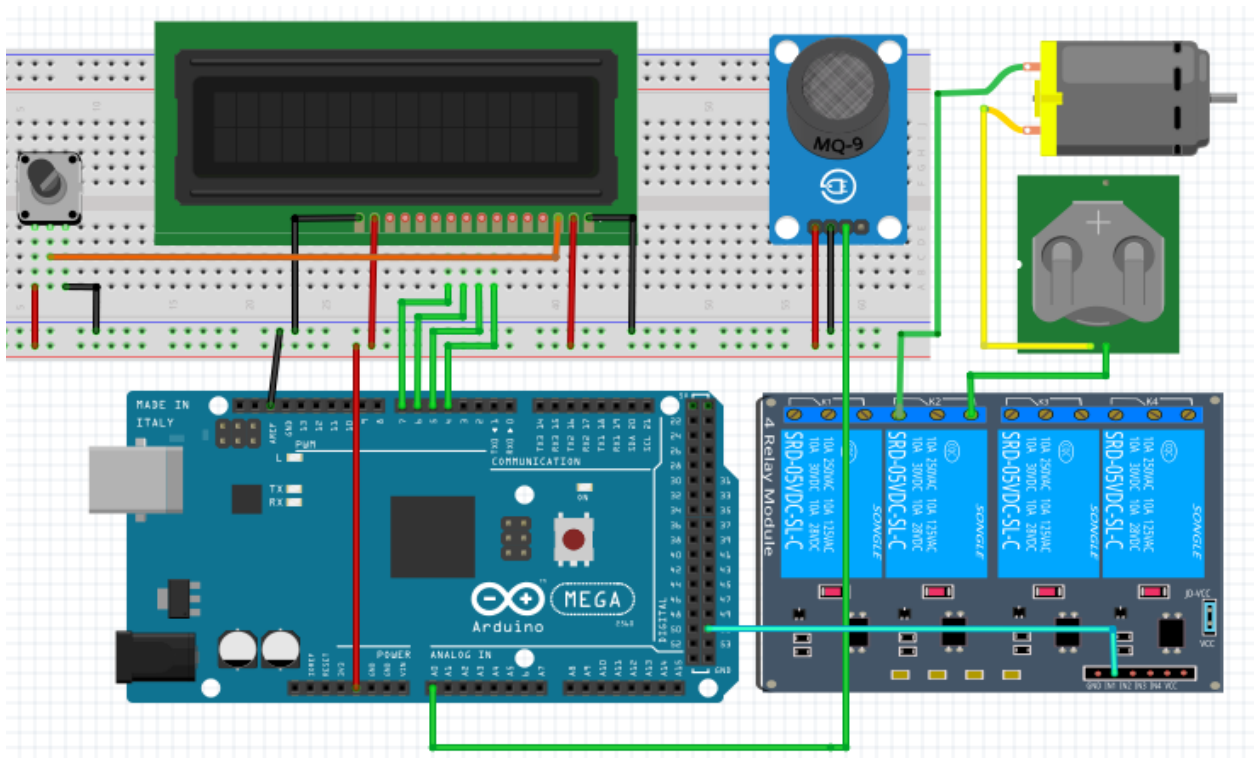
Hình 3.3. Mạch cảm biến MQ 135 đo nồng độ khí độc hại.

THÔNG SỐ KỸ THUẬT

- Điện áp nguồn: $\leq 24\text{VDC}$
- Điện áp của heater: $5\text{V} \pm 0.1 \text{ AC/DC}$
- Điện trở tải: thay đổi được ($2\text{k}\Omega - 47\text{k}\Omega$)
- Điện trở của heater: $33\Omega \pm 5\%$
- Công suất tiêu thụ của heater: ít hơn 800mW
- Khoảng phát hiện: 10 – 300 ppm NH₃, 10 – 1000 ppm Benzene, 10 – 300 Alcol
- Kích thước: 32mm*20mm

Cảm biến có thể kết nối với Kit Arduino qua lối ra tín hiệu tương tự hoặc tín hiệu số. Trong thiết kế này em sử dụng kết nối qua lối ra tương tự để sử dụng bộ biến đổi ADC của mạch Arduino.

Sơ đồ đấu nối mạch đo và hiển thị tại chỗ được đưa ra trên hình 3.4.



Hình 3.4. Sơ đồ hệ thống đo nồng độ khí độc hại
hiển thị kết quả tại chỗ qua màn LCD.

Hoạt động của hệ thống như sau: Tín hiệu thu được từ cảm biến được đưa vào chân A0 của bộ Mega 2560. Giá trị của cảm biến sẽ được hiển thị trên màn hình LCD nếu cần quan sát tại chỗ. Trong trường hợp nồng độ khí độc hại là khí GAS vượt ngưỡng cho phép (trong thiết kế đặt giá trị ngưỡng là 200) thì bộ xử lý trung tâm ra

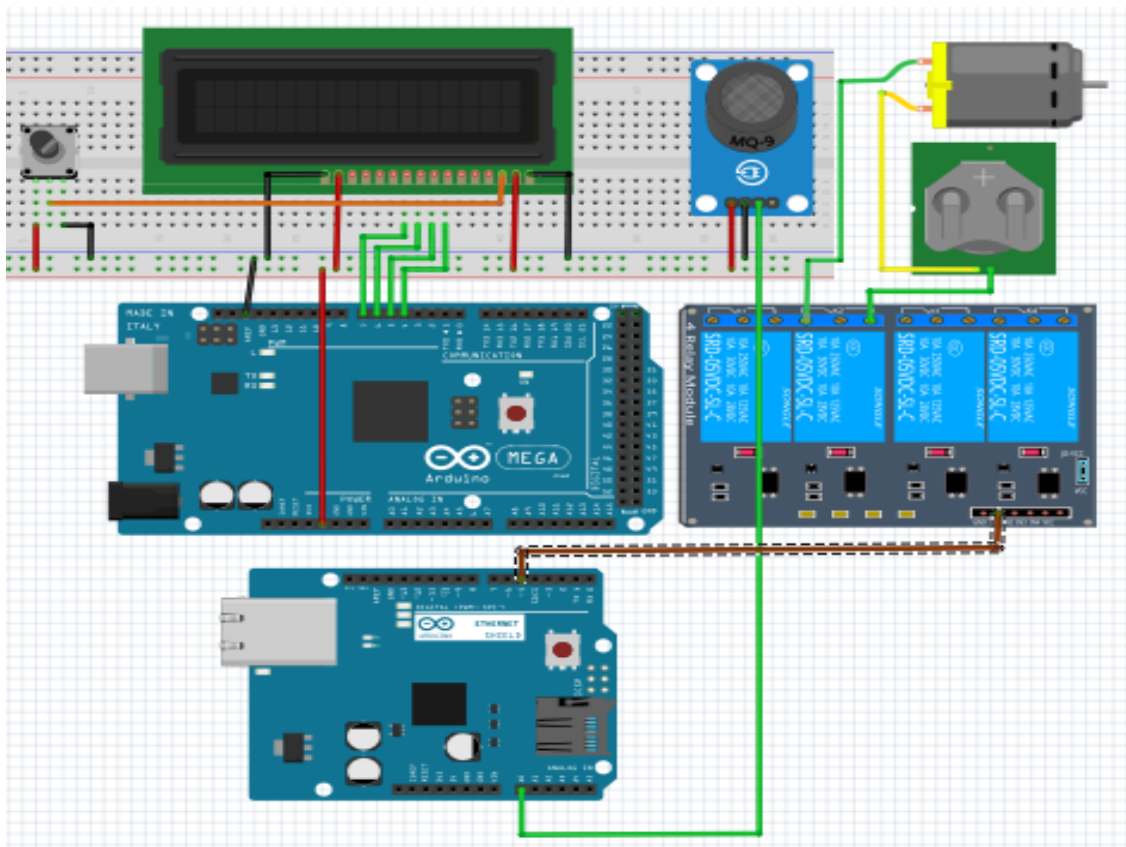
tín hiệu điều khiển cho Relay chuyển chế độ và cấp nguồn điện áp 220V AC cho hệ thống quạt thông gió. Do đó sẽ làm mới lại môi trường không khí đang cần kiểm tra giám sát. Khi chất lượng không khí tốt hơn, nồng độ khí độc hại giảm đi. Giá trị cảm biến thu được nhỏ hơn ngưỡng khi bộ xử lý trung tâm ra tín hiệu cho Relay chuyển sang chế độ ngắt và như vậy hệ thống thông gió ngừng hoạt động. Hệ thống hoạt động hoàn toàn tự động theo chương trình được nạp cho Arduino Mega 2560.

3.3. Giám sát độ ẩm qua Internet

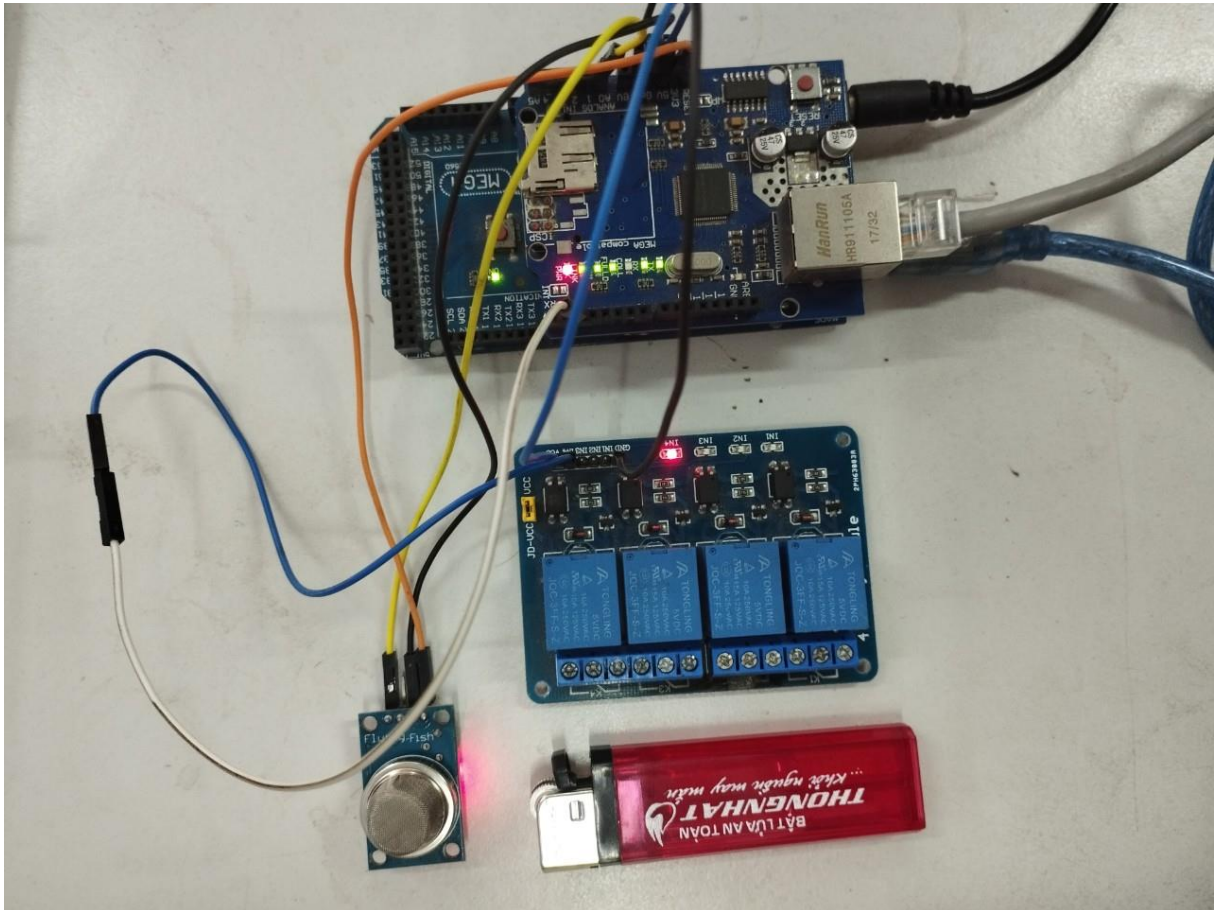
Trong đề án này em sẽ sử dụng Arduino Ethernet W5100 để thực hiện một ứng dụng về đo tín hiệu từ cảm biến sau đó truyền tín hiệu tự động về người giám sát qua hệ thống mạng Internet. Có thể coi đó là một hệ thống IoT đơn giản.

Yêu cầu của người dùng là đo được nồng độ khí độc hại thì thực hiện lệnh cho bật tự động hệ thống thông gió. Khi chất lượng không khí môi trường trở nên tốt hơn, nồng độ khí độc hại giảm dưới giá trị ngưỡng thì tự động tắt hệ thống quạt thông gió. Giá trị của

Để thiết kế hệ thống, sử dụng Arduino Ethernet W5100 vừa làm nhiệm vụ kết nối internet qua cổng RJ45 vừa làm nhiệm vụ nhận và xử lý tín hiệu đo được từ cảm biến chất lượng không khí MQ135. Sơ đồ phân cứng được thiết kế như hình 3.6



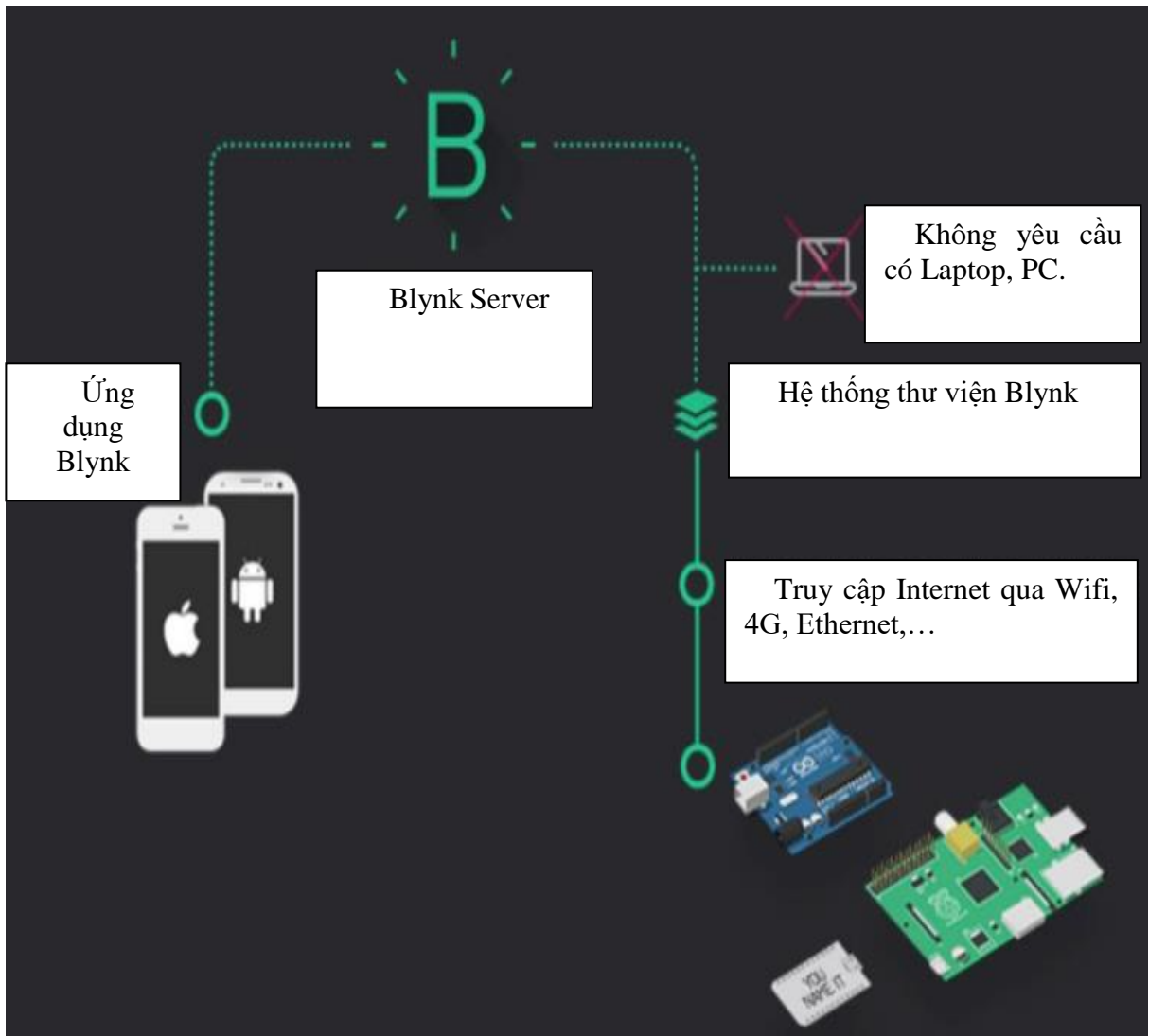
Hình 3.6. Sơ đồ phân cứng hệ thống giám sát nồng độ khí độc hại qua Internet.



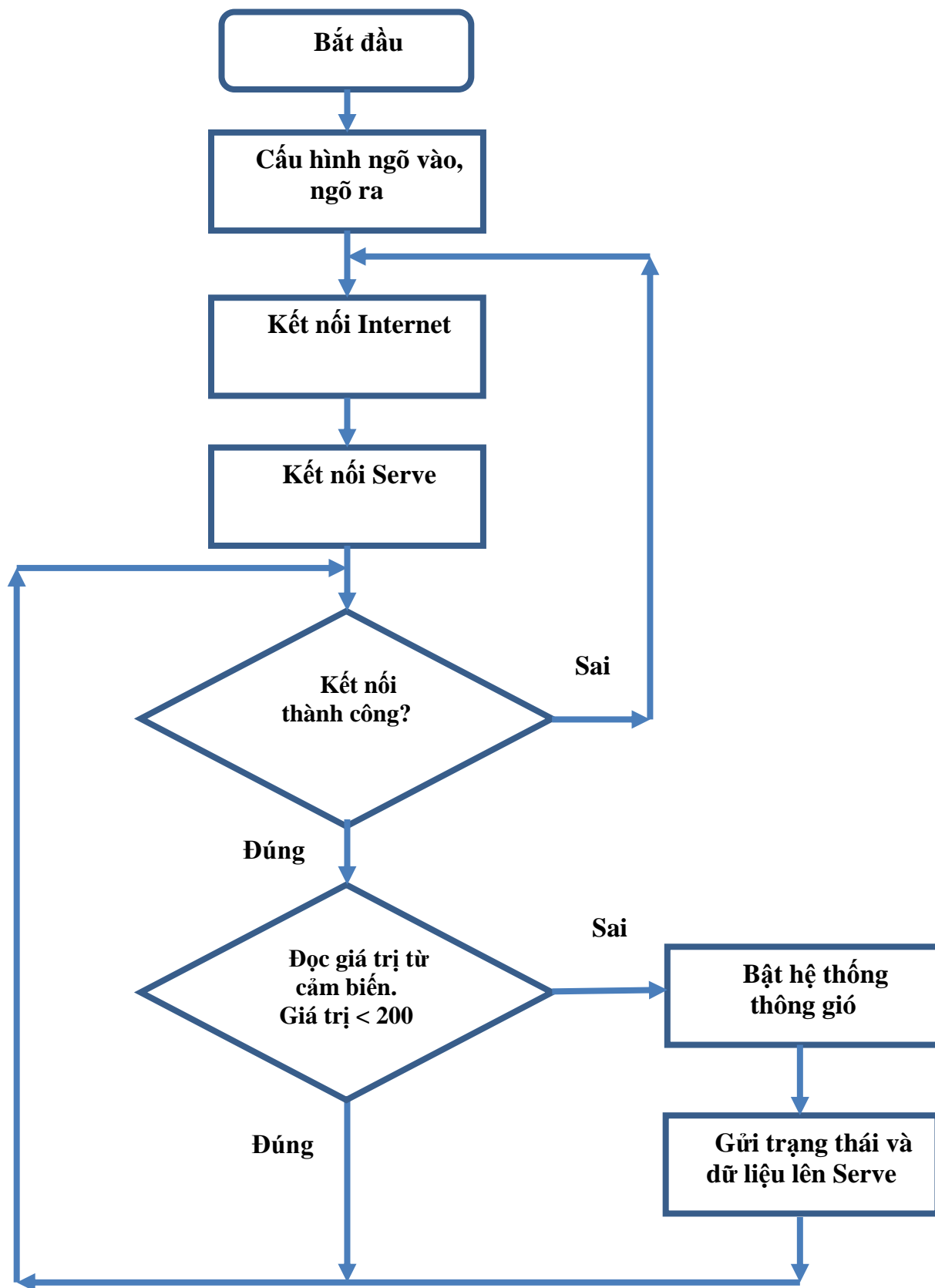
Hình 3.7. Hình ảnh mô hình thực tế.

Để thực hiện đưa tín hiệu đo nồng độ khí độc hại và các thao tác điều khiển quạt thông gió từ xa qua Internet có thể dùng nhiều công cụ hỗ trợ như Blynk, Google Assistant,... Trong nghiên cứu này sử dụng WEB Serve Blynk để kết nối điện thoại dùng để giám sát thiết bị kết nối qua mạng Internet. Sơ đồ khối hệ thống kết nối như hình 3.8 dưới đây.

Lưu đồ thuật toán đo và giám sát nồng độ khí GAS đưa ra trên hình 3.9. Theo lưu đồ thuật toán, ban đầu chúng ta thực hiện các lệnh cấu hình ngõ vào/ra của KIT Arduino được sử dụng. Sau đó đưa các thông số kết nối mạng vào cho KIT. Bao gồm các thông số chính là: mã xác thực quyền trên Blynk, tên hệ thống Wifi và mật khẩu được dùng. Việc kết nối thành công bản chất sẽ được thông báo trên điện thoại được cài ứng dụng Blynk. Khi hệ thống kết nối thành công, chúng ta thực hiện đọc giá trị từ cảm biến và kiểm tra xem giá trị đó có vượt quá nồng độ cho phép không. Nếu nồng độ khí độc hại vượt quá giá trị ngưỡng (>200) thì thực hiện lệnh cho phép bật hệ thống quạt thông gió một cách tự động. Nếu nồng độ khí GAS nhỏ hơn giá trị ngưỡng thì không làm gì cả. Chương trình được thực hiện khép kín như vậy. Code chi tiết của chương trình được tác giả trình bày ở phụ lục của đồ án.



Hình 3.8. Sơ đồ khối kết nối qua Blynk Server.

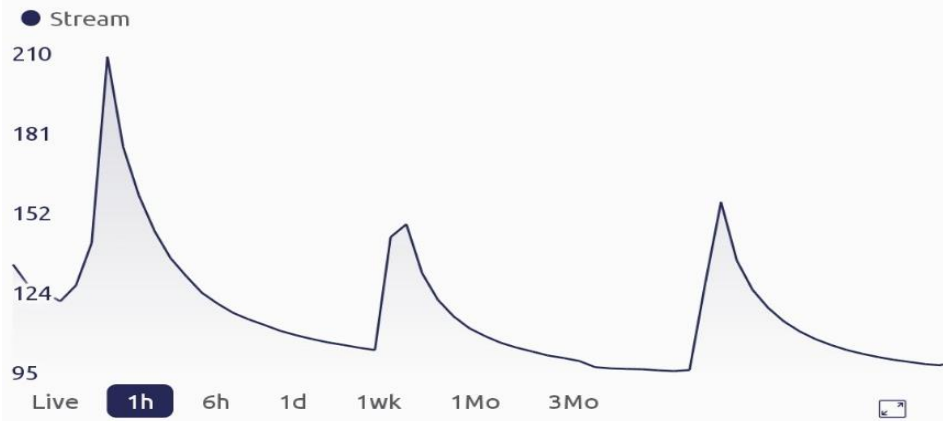


Hình 3.9. Lưu đồ thuật toán đo và giám sát độ ẩm.

Một số hình ảnh ví dụ của việc giám sát nồng độ khí độc hại và điều khiển hệ thống quạt thông gió được đưa ra ở dưới đây.

160

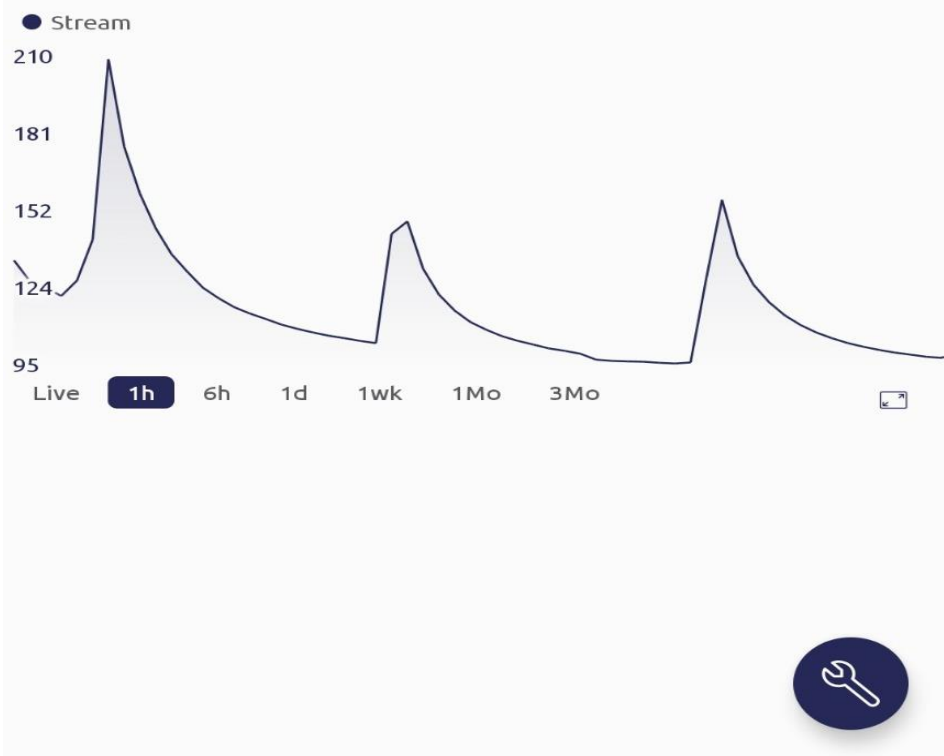
```
HELLO  
value la:161
```



Hình 3.10. Nồng độ khí GAS ở mức thấp.

216

```
HELLO  
value 1a:216
```



Hình 3.11. Nồng độ khí GAS ở mức cao.

Kết luận

Sau thời gian làm đồ án tốt nghiệp, em đã thực hiện tìm hiểu được tổng quan về hệ thống Internet kết nối vạn vật IoT, tìm hiểu được hoạt động của các loại cảm biến độ nồng độ khí độc hại, khí GAS và một số Kit Arduino thường dùng trong thực tế. Em cũng đã nghiên cứu tìm hiểu được ứng dụng của ứng dụng Blynk IoT vào việc xây dựng hệ thống IoT. Từ đó, em đã thực hiện thiết kế và xây dựng thành công hệ thống đo và giám sát nồng độ khí độc hại trong môi trường qua Internet.

Em xin chân thành cảm ơn các thầy trong Khoa Điện – Điện tử đã luôn nhiệt tình hỗ trợ và giảng dạy chúng em trong suốt những năm tháng học tập tại trường. Em xin cảm ơn thầy TS Đoàn Hữu Chúc đã hướng dẫn và chỉ bảo em trong thời gian làm đồ án và giúp em hoàn thành được đồ án này.

Sinh viên thực hiện

Kim Văn Dinh

Phụ lục

```
// Chương trình giám sát nồng độ khí GAS
// và điều khiển thiết bị

#define BLYNK_TEMPLATE_ID "TMPLJ-Xfh_iA"
#define BLYNK_TEMPLATE_NAME "EthernetDinh"
#define BLYNK_AUTH_TOKEN "S_0iv8JjpMUL9Q-HxHs2ad_FyZFMtA7R"

#include <SPI.h>
#include <Ethernet.h>
#include <BlynkSimpleEthernet.h>

byte mac[] = {0x3C, 0xDA, 0x2A, 0xD2, 0x39, 0x97};
//byte ip[] = { 192, 168, 1, 99 };
//byte gateway[] = { 192, 168, 1, 1 };
//byte subnet[] = { 255, 255, 255, 0 };
IPAddress ip(192, 168, 1, 99);
EthernetServer server(80);
char auth[] = "S_0iv8JjpMUL9Q-HxHs2ad_FyZFMtA7R";
int CB = A0;
int button;
int relay = 2;
WidgetLCD cambien(V4);
void setup()
{
  Serial.begin(9600);
  Blynk.begin(auth);
  pinMode(relay, OUTPUT);
  pinMode(A0, INPUT);
}
void loop()
{
  Blynk.run();
  CB = analogRead(A0);
  int giatrithuc = map(CB, 0, 1023, 1, 1024);
```

```
//int giatri = 1024 - giatrithuc;
int giatri=giatrithuc;
Blynk.virtualWrite(V2, giatri);
if(giatri > 200){
digitalWrite(relay,HIGH);
}
else{
    digitalWrite(relay,LOW);
}
cambien.print(0, 0, "HELLO");
cambien.print( 0, 1, "value la:" + String (giatri));
}
```

TÀI LIỆU THAM KHẢO

1. Giáo trình Kỹ Thuật Lập Trình C Căn Bản Và nâng cao - Phạm Văn Ất
2. Adeel Javed, “Building Arduino Projects for the Internet of Things”, Apress, 2016.
3. Hà Quang Phúc, “Lập Trình Điều Khiển Trên Arduino Cho Hệ Vạn Vật Kết Nối (IoT)”, NXB Thanh Niên, 2020.
4. <http://arduino.vn/>
5. <https://www.arduino.cc/>
6. Jack Purdum, “Beginning C for Arduino”, Apress, 2012.
7. Lê Mỹ Hà, Phạm Quang Huy, “Lập trình IoT với Arduino”, NXB Thanh niên, 2019.
8. Marco Schwartz, “Internet of Things with Arduino Cookbook”, Packt Publishing Ltd, 2016.
9. Nguyễn Vũ Quỳnh, Phạm Quang Huy, “Giáo trình đo lường cảm biến (Lý thuyết – Thực hành)”, NXB Thanh Niên, 2020.