

TRƯỜNG ĐH CÔNG NGHIỆP THỰC PHẨM TP. HCM

KHOA CÔNG NGHỆ ĐIỆN – ĐIỆN TỬ

BỘ MÔN TỰ ĐỘNG HÓA



ĐỒ ÁN HỌC PHẦN 2

**MÔ HÌNH HỆ THỐNG BÁO CHÁY DỪNG
MODULE SIM**

GVHD: ThS. TRẦN HOÀN

SVTH: Nguyễn Thành Công

MSSV: 2032180110

TP. HỒ CHÍ MINH, tháng 7 năm 2021

TRƯỜNG ĐH CÔNG NGHIỆP THỰC PHẨM TP. HCM

KHOA CÔNG NGHỆ ĐIỆN – ĐIỆN TỬ

BỘ MÔN TỰ ĐỘNG HÓA



ĐỒ ÁN HỌC PHẦN 2

MÔ HÌNH HỆ THỐNG BÁO CHÁY DỪNG

MODULE SIM

GVHD: ThS. TRẦN HOÀN

SVTH: Nguyễn Thành Công

MSSV: 2032180110

TP. HỒ CHÍ MINH, tháng 7 năm 2021

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THỰC PHẨM TP.
HCM
KHOA CN ĐIỆN – ĐIỆN TỬ
BỘ MÔN: TỰ ĐỘNG HÓA

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc

TP. HCM, ngày.....tháng.....năm.....

NHẬN XÉT ĐỒ ÁN HỌC PHẦN 2 CỦA GIẢNG VIÊN HƯỚNG DẪN

Tên đồ án:

MÔ HÌNH HỆ THỐNG BÁO CHÁY DÙNG MODULE SIM

Sinh viên thực hiện:

Nguyễn Thành Công

2032180110

Giảng viên hướng dẫn:

ThS. TRẦN HOÀN

Đánh giá Đồ án

1. Về cuốn báo cáo:

Số trang	_____	Số chương	_____
Số bảng số liệu	_____	Số hình vẽ	_____
Số tài liệu tham khảo	_____	Sản phẩm	_____

Một số nhận xét về hình thức cuốn báo cáo:

2. Về nội dung đồ án:

3. Về tính ứng dụng:

4. Về thái độ làm việc của sinh viên:

Đánh giá chung:

Điểm từng sinh viên:

Nguyễn Thành Công:...../10

Người nhận xét

(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

TP. Hồ Chí Minh, ngày ... tháng 09 năm 2021

Tác giả

Nguyễn Thành Công

TP. HCM, ngày....tháng.....năm.....

ĐỀ CƯƠNG CHI TIẾT

TÊN ĐỒ ÁN: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG BÁO CHÁY TỰ ĐỘNG	
Giảng viên hướng dẫn: ThS.Trần Hoàn	
Thời gian thực hiện: Từ ngày .../.../201... đến ngày .../.../201...	
Sinh viên thực hiện: Nguyễn Thành Công	
Nội dung đề tài: <ul style="list-style-type: none">- Tìm hiểu về arduino; module Sim 800A; cảm biến MQ2, DTH11- Đọc các giá trị khí gas, nhiệt độ, độ ẩm từ cảm biến MQ2, DTH11- Hiển thị lên LCD giá trị đọc được của cảm biến- Cảnh báo thông qua led;còi; gọi, nhắn tin về Sim khi các giá trị của cảm biến vượt ngưỡng cho phép .	
Kế hoạch thực hiện: <ul style="list-style-type: none">- Từ ngày .../.../201... đến ngày .../.../201...: (hoặc tháng...): thực hiện.....- Từ ngày .../.../201... đến ngày .../.../201...: (hoặc tháng...): thực hiện.....:- Từ ngày .../.../201... đến ngày .../.../201...: (hoặc tháng...): thực hiện.....:- Từ ngày .../.../201... đến ngày .../.../201...: (hoặc tháng...): thực hiện.....:	
Xác nhận của giảng viên hướng dẫn	TP. HCM, ngày....thángnăm..... Sinh viên Nguyễn Thành Công

MỤC LỤC

MỤC LỤC	i
DANH MỤC KÝ HIỆU, CỤM TỪ VIẾT TẮT.....	iii
DANH MỤC HÌNH ẢNH	iv
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....	1
1.1 Đặt vấn đề.....	1
1.2 Mục tiêu đề tài	1
1.3 Phương pháp nghiên cứu	1
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	2
2.1 Arduino	2
2.1.1 Giới thiệu.....	2
2.1.2 Thông số kỹ thuật	3
2.2 Màn hình LCD 16x2.....	6
2.2.1 Giới thiệu.....	6
2.2.2 Thông số kỹ thuật.....	7
2.2.3 Chức năng các chân.....	7
2.3 Module giao tiếp I2C với LCD.....	8
2.3.1 Giới thiệu.....	8
2.3.2 Thông số kỹ thuật.....	9
2.4 Cảm biến nhiệt độ, độ ẩm DHT11.....	10
2.4.1 Giới thiệu.....	10
2.4.2 Thông số kỹ thuật.....	10
2.4.3 Nguyên lý hoạt động	10
2.5 Cảm biến khí gas MQ-2	11
2.5.1 Giới thiệu.....	11
2.5.2 thông số kỹ thuật	11
2.6 Module SIM 800A.....	12
2.6.1 Giới thiệu.....	12

2.6.2 Thông số kỹ thuật.....	12
2.6.3 Chức năng các chân.....	13
2.6.4 Tập lệnh AT của module SIM 800A.....	13
2.7 Mạch giảm áp DC LM2596.....	15
2.7.1 Giới thiệu.....	15
2.7.2 Thông số kỹ thuật.....	15
CHƯƠNG 3: CƠ SỞ THỰC HIỆN.....	16
3.1 Thiết kế sơ đồ khối của hệ thống.....	16
3.2 Thiết kế lưu đồ giải thuật.....	17
3.3 Thiết kế sơ đồ kết nối của hệ thống.....	18
CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM	19
4.1 Lượng khí gas vượt ngưỡng cho phép.....	19
4.2 Nhiệt độ vượt ngưỡng cho phép.....	19
4.3 Nhiệt độ và khí gas cùng vượt ngưỡng cho phép.....	21
CHƯƠNG 5: KẾT LUẬN VÀ ĐỊNH HƯỚNG ĐỀ TÀI	23
5.1 Kết quả đạt được.....	23
5.2 Hạn chế	23
5.3 Hướng phát triển của đề tài	23
PHỤ LỤC	24
TÀI LIỆU THAM KHẢO	32

DANH MỤC KÝ HIỆU, CỤM TỪ VIẾT TẮT

KÝ HIỆU	THUẬT NGỮ
LCD	liquid crystal display
DC	Direct curent
I/O	Input/output
IDE	Integrated Development Environment
EEPROM	Electrically Erasable Programmable Read-Only Memory
SRAM	Static random-access memory
PWM	Pulse Width Modulation
GSM	Grams per square Meter
GPRS	General Packet Radio Service
RAM	Random Access Memory

DANH MỤC HÌNH ẢNH

Hình 2.1. Cấu trúc phần cứng Arduino	3
Hình 2.2. Sơ đồ chân trong ATmega 328.....	4
Hình 2.3. Sơ đồ khối cấu trúc bên trong ATmega 328.....	5
Hình 2.4. Sơ đồ cấu trúc CPU trong ATmega 328.....	6
Hình 2.5. LCD 16x2	7
Hình 2.6. Bảng chức năng của các chân LCD.....	8
Hình 2.7. Module I2C và các chân chức năng	9
Hình 2.8. Module cảm biến nhiệt độ, độ ẩm DHT11	10
Hình 2.9. Sơ đồ kết nối chân DHT11	11
Hình 2.10. Module SIM800A.....	12
Hình 3.1. Sơ đồ khối của mạch	16
Hình 3.2. Lưu đồ giải thuật.....	17
Hình 3.3. Sơ đồ kết nối của hệ thống	18
Hình 4.1. Khí gas vượt ngưỡng cho phép.....	19
Hình 4.2. Nhiệt độ vượt ngưỡng cho phép.....	20
Hình 4.3. Nhiệt độ vượt ngưỡng cho phép.....	20
Hình 4.4. Nhiệt độ và khí gas vượt ngưỡng cho phép.....	21
Hình 4.5. Nhiệt độ và khí gas vượt ngưỡng cho phép.....	21
Hình 4.6. Nhiệt độ và khí gas vượt ngưỡng cho phép.....	22

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1 Đặt vấn đề

Hiện nay với sự phát triển của nước ta với những máy móc công nghệ tiên tiến đã giúp phần nào giúp cho nhu cầu người dùng dễ dàng hơn thì song song theo đó thì chúng ta không tránh được những nguy cơ cháy nổ ngoài ý muốn và đã làm thiệt hại to lớn về tài sản rất nhiều. Có thể do chúng ta chưa có phòng sự phòng tránh kịp thời đúng lúc để có thể làm giảm phần nào về thiệt hại tài sản. Từ lý do đó nên em chọn đề tài “Hệ thống báo cháy tự động qua điện thoại”.

1.2 Mục tiêu đề tài

- Giám sát, điều khiển hoạt động của mạch báo cháy.
- Đo nhiệt độ, độ ẩm nơi gắn mạch.
- Hiện thị trạng thái hệ thống qua LCD 16x2.
- Cảnh báo khi nhiệt độ tăng cao.
- Cảnh báo khi khí gas vượt ngưỡng.
- Gửi được tin nhắn về điện thoại thông qua module SIM800A.
- Gọi được về điện thoại thông qua module SIM800A.

1.3 Phương pháp nghiên cứu

Dựa vào datasheet của từng linh kiện cho ta biết được chức năng nhiệm vụ của từng chân. Ngoài ra em còn tham khảo nguồn thông tin, bài viết, video trên các kênh phương tiện thông tin. Từ đó, từng bước thực hiện và trong quá trình thực hiện có lỗi, khó khăn xuất hiện em thì phân tích lỗi và giải quyết.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Arduino

2.1.1 Giới thiệu

Board Arduino, nơi thực thi các chương trình lập trình. Các board này có thể điều khiển hoặc đáp trả các tín hiệu điện, vì vậy các thành phần được ghép trực tiếp vào nó nhằm tương tác với thế giới thực để cảm nhận và truyền thông. Ví dụ các cảm biến bao gồm các thiết bị chuyển mạch, cảm biến siêu âm, gia tốc. Các thiết bị truyền động bao gồm đèn, motor, loa và các thiết bị hiển thị.

Có rất nhiều ứng dụng sử dụng Arduino để điều khiển. Arduino có rất nhiều module, mỗi module được phát triển cho một ứng dụng. Về mặt chức năng, các bo mạch Arduino được chia thành hai loại: loại bo mạch chính có chip Atmega và loại mở rộng thêm chức năng cho bo mạch chính. Các bo mạch chính về cơ bản là giống nhau về chức năng, tuy nhiên về mặt cấu hình như số lượng I/O, dung lượng bộ nhớ, hay kích thước có sự khác nhau. Một số bo mạch có trang bị thêm các tính năng kết nối như Ethernet và Bluetooth. Các bo mở rộng chủ yếu mở rộng thêm một số tính năng cho bo mạch chính ví dụ như tính năng kết nối Ethernet, Wireless, điều khiển động cơ.

Arduino Uno là một bo mạch vi điều khiển dựa trên chip ATmega168 hoặc ATmega 328. Cấu trúc chung bao gồm:

- 14 chân vào ra bằng tín hiệu số, trong đó có 6 chân có thể sử dụng để điều chế độ rộng xung.
- Có 6 chân đầu vào tín hiệu tương tự cho phép chúng ta kết nối với các bộ cảm biến bên ngoài để thu thập số liệu.
- Sử dụng một dao động thạch anh tần số dao động 16MHz.
- Có một cổng kết nối bằng chuẩn USB để chúng ta nạp chương trình vào bo mạch và một chân cấp nguồn cho mạch, một nút reset.
- Nó chứa tất cả mọi thứ cần thiết để hỗ trợ các vi điều khiển, nguồn cung cấp cho Arduino có thể là từ máy tính thông qua cổng USB hoặc là từ bộ nguồn

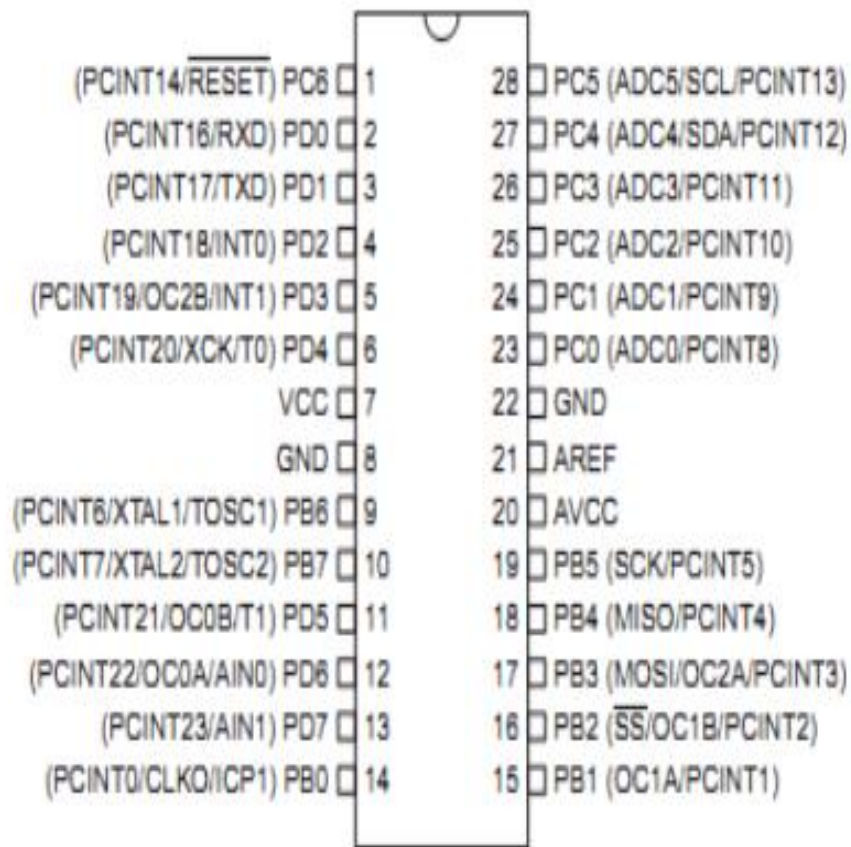
chuyên dụng được biến đổi từ xoay chiều sang một chiều hoặc là nguồn lấy từ pin.



Hình 2.1. Cấu trúc phần cứng Arduino

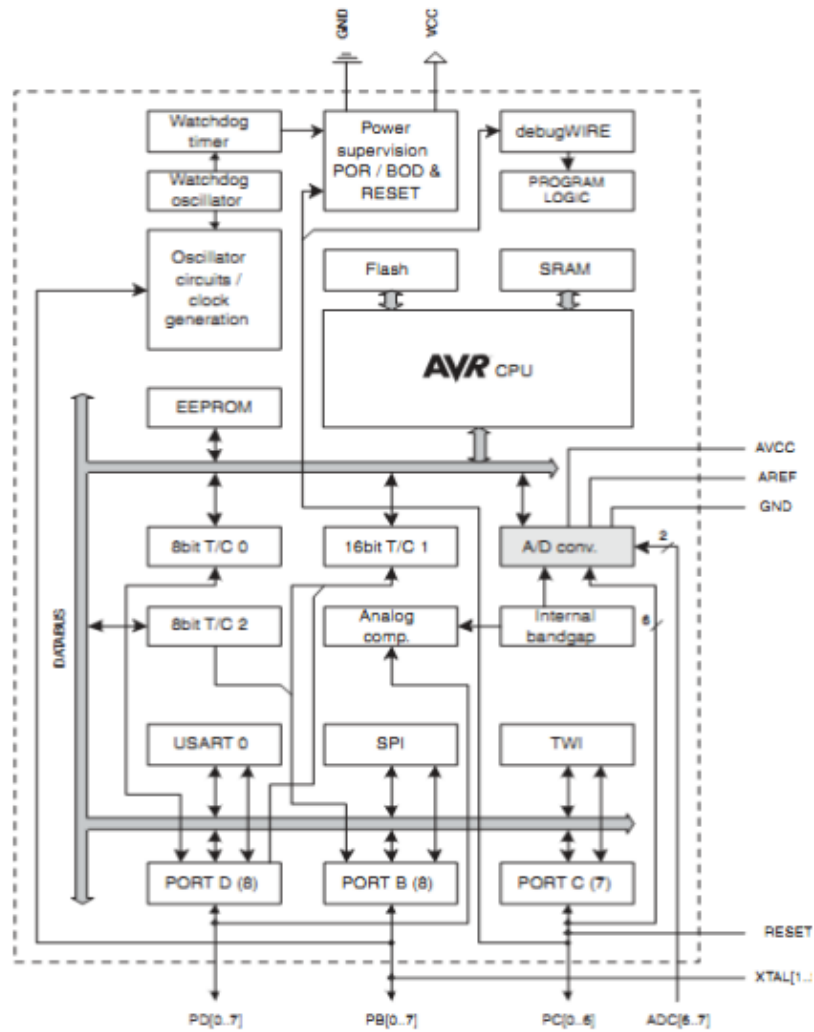
2.1.2 Thông số kỹ thuật

- Khối xử lý trung tâm là vi điều khiển Atmega328.
- Điện áp hoạt động 5V.
- Điện áp đầu vào khuyến nghị là 5-12V.
- Điện áp đầu vào giới hạn 6-20V.
- Dòng điện một chiều trên các chân vào ra là 40mA.
- Dòng điện một chiều cho chân 3.3V là 50mA.
- Clock Speed 16 MHz.
- Flash Memory 16 Kb (ATmega 168) hoặc 32 Kb (ATmega 328), SRAM 1 Kb (ATmega 168) hoặc 2 Kb (ATmega 328), EEPROM 512 bytes (ATmega 168) hoặc 1 Kb (ATmega 328).
- ❖ Khối xử lý trung tâm : Trong bo mạch Arduino IC đóng vai trò xử lý trung tâm là Atmega328 cấu trúc sơ đồ chân của nó như sau:



Hình 2.2. Sơ đồ chân trong ATmega 328

- Chân VCC (chân số 7): Chân cung cấp điện áp dương nguồn 5V.
- Chân GND (chân số 8): Chân đất chung.
- Chân AREF (chân 21): Là chân tham chiếu để chuyển đổi tín hiệu tương tự sang số.
- Chân AVCC (chân 20): Chân cung cấp điện áp cho quá trình chuyển đổi ADC.
- Cổng B (chân 14 - chân 19, chân 9, chân 10): Bao gồm có 8 chân I/O từ (PB0÷PB7).
- Cổng C (chân 23 – chân 28, chân 1): Bao gồm có 7 chân I/O từ (PC0÷PC6) trong đó chân PC6 (chân số 1) làm chân reset.
- Cổng D (chân 2 – chân 6, chân 11 – chân 13): Bao gồm có 8 chân I/O từ chân (PD0÷PD7).



Hình 2.3. Sơ đồ khối cấu trúc bên trong ATmega 328

❖ Khối xử lý trung tâm trong IC ATmega 328 như sau:

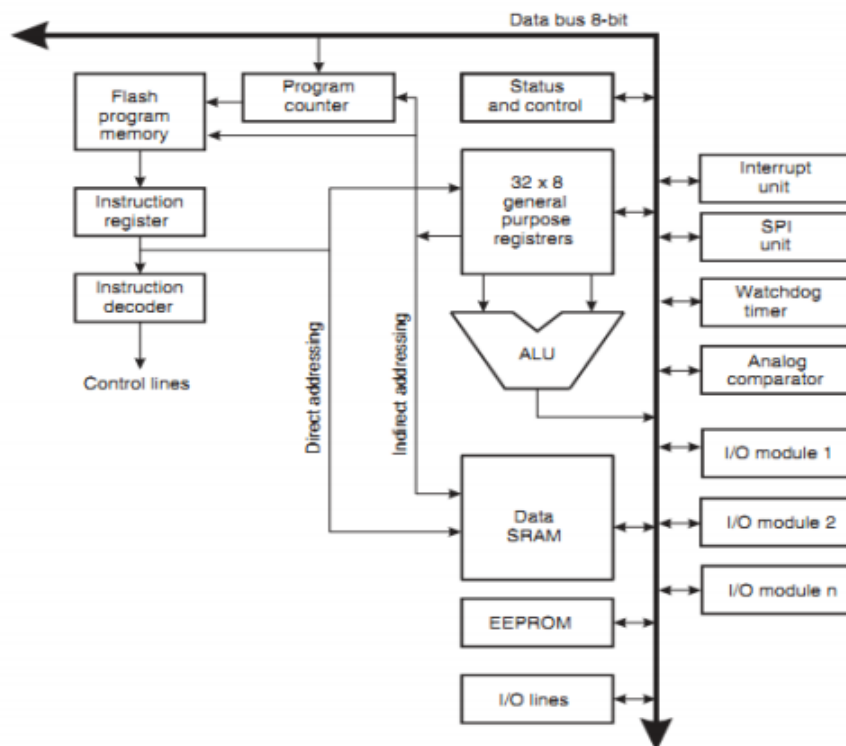
Đây là kiến trúc chung trong lõi AVR nói chung. Chức năng chính của lõi CPU là để đảm bảo thực hiện chương trình chính xác. CPU do đó phải có khả năng truy cập nhanh, thực hiện các tính toán, thiết bị ngoại vi điều khiển và xử lý ngắt. Để tối đa hóa hiệu suất, AVR sử dụng một kiến trúc Harvard và đường bus riêng biệt cho chương trình và dữ liệu. Hướng truyền dữ liệu trong bộ nhớ chương trình thực hiện với một tốc độ nhất định.

❖ Nguồn nuôi

Arduino có thể được hỗ trợ thông qua kết nối USB hoặc với một nguồn cung cấp điện bên ngoài. Các nguồn năng lượng được lựa chọn tự động. Hệ thống vi điều khiển có thể hoạt động bằng một nguồn cung cấp bên ngoài từ 6V đến 20V. Nên

cung cấp với ít hơn 7V, tuy nhiên pin 5V có thể cung cấp ít hơn 5V và hệ thống vi điều khiển có thể không ổn định. Nếu sử dụng nhiều hơn 12V điều chỉnh điện áp có thể quá nóng. Phạm vi khuyến nghị là 7V đến 12V.

- Chân Vin: Điện áp đầu vào Arduino khi chúng ta dùng nguồn điện bên ngoài. Chúng ta có thể cung cấp nguồn thông qua chân này.
- Chân 5V: Cung cấp nguồn vi điều khiển và các bộ phận khác trên bo mạch và cung cấp nguồn cho các thiết bị ngoại vi khi kết nối tới bo mạch.
- Chân 3V3: Cung cấp nguồn cho các thiết bị cảm biến.
- Chân GND : Chân nối đất.



Hình 2.4. Sơ đồ cấu trúc CPU trong ATmega 328

2.2 Màn hình LCD 16x2

2.2.1 Giới thiệu

Hiện giờ, thiết bị hiển thị LCD 1602 (Liquid Crystal Display) được dùng trong rất nhiều các ứng dụng của VDK. LCD 1602 có rất nhiều ưu điểm so với các dạng

hiển thị khác như: khả năng hiển thị ký tự đa dạng (ký tự đồ họa, chữ, số,); đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau dễ dàng , tiêu tốn rất ít tài nguyên hệ thống, giá thành rẻ,...



Hình 2.5. LCD 16x2

2.2.2 Thông số kỹ thuật

- Điện áp MAX : 7V
- Điện áp MIN : - 0,3V
- Điện áp ra mức thấp : <0.4V
- Điện áp ra mức cao : > 2.4
- Hoạt động ổn định : 2.7-5.5V
- Dòng điện cấp nguồn : 350uA - 600uA
- Nhiệt độ hoạt động : - 30 - 75 độ C

2.2.3 Chức năng các chân

Trong 16 chân của LCD được chia ra làm 3 dạng tín hiệu như sau:

- Các chân cấp nguồn: Chân số 1 là chân nối mass (0V), chân thứ 2 là Vdd nối với nguồn +5V. Chân thứ 3 dùng để chỉnh contrast thường nối với biến trở.
- Các chân điều khiển: Chân số 4 là chân RS dùng để điều khiển lựa chọn thanh ghi. Chân R/W dùng để điều khiển quá trình đọc và ghi.
- Các chân dữ liệu D7÷D0: Chân số 7 đến chân số 14 là 8 chân dùng để trao đổi dữ liệu giữa thiết bị điều khiển và LCD.

Chân	Ký hiệu	I/O	Mô tả
1	VSS	Nguồn	GND
2	VDD	Nguồn	+5V
3	VEE	Điện áp	Đ
4	RS	INPUT	Chân chọn thanh ghi (Register select)
5	RW	INPUT	Chân chọn chế độ đọc/ghi (Read/Write)
6	E	INPUT	Chân cho phép (Enable)
7	DB0	I/O	DATA LSB
8	DB1	I/O	DATA
9	DB2	I/O	DATA
10	DB3	I/O	DATA
11	DB4	I/O	DATA
12	DB5	I/O	DATA
13	DB6	I/O	DATA
14	DB7	I/O	DATA
15	A	I	Nguồn dương +5V
16	K	I	GND

Hình 2.6. Bảng chức năng của các chân LCD

2.3 Module giao tiếp I2C với LCD

2.3.1 Giới thiệu

LCD có quá nhiều chân gây khó khăn trong quá trình kết nối và chiếm dụng nhiều chân của vi điều khiển. Để khắc phục tình trạng đó thì ta có thể sử dụng

module chuyển đổi I2C cho LCD. Thay vì sử dụng tối thiểu 6 chân của vi điều khiển để kết nối với LCD (RS, EN, D7, D6, D5 và D4) thì với module chuyển đổi, chỉ cần sử dụng 2 chân (SCL, SDA) để kết nối. Module chuyển đổi I2C hỗ trợ các loại LCD sử dụng driver HD44780 (LCD 1602, LCD 2004, ...), kết nối với vi điều khiển thông qua giao tiếp I2C, tương thích với hầu hết các vi điều khiển hiện nay.



Hình 2.7. Module I2C và các chân chức năng

Ưu điểm :

- Tiết kiệm chân cho vi điều khiển.
- Dễ dàng kết nối với LCD.

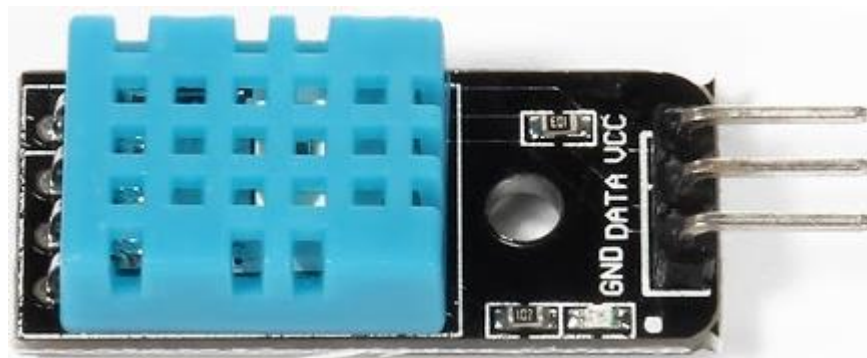
2.3.2 Thông số kỹ thuật

- Điện áp hoạt động: 2.5-6V DC
- Hỗ trợ màn hình: LCD1602,1604,2004 (driver HD44780)
- Giao tiếp: I2C
- Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2)
- Kích thước: 41.5mm(L)x19mm(W)x15.3mm(H)
- Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt
- Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD

2.4 Cảm biến nhiệt độ, độ ẩm DHT11

2.4.1 Giới thiệu

DHT11 Là cảm biến rất thông dụng hiện nay vì chi phí rẻ và rất dễ lấy dữ liệu thông qua giao tiếp 1-wire (giao tiếp digital 1-wire truyền dữ liệu duy nhất). Cảm biến được tích hợp bộ tiền xử lý tín hiệu giúp dữ liệu nhận về được chính xác mà không cần phải qua bất kỳ tính toán nào.



Hình 2.8. Module cảm biến nhiệt độ, độ ẩm DHT11

2.4.2 Thông số kỹ thuật

- Điện áp hoạt động: 3-5.5VDC
- Chuẩn giao tiếp: TTL, 1 wire.
- Khoảng đo độ ẩm: 20%-80% RH sai số $\pm 5\%RH$
- Khoảng đo nhiệt độ: 0-50°C sai số $\pm 2^\circ C$
- Tần số lấy mẫu tối đa 1Hz (1 giây / lần)
- Kích thước: 28mm x 12mm x 10mm

2.4.3 Nguyên lý hoạt động

Để có thể giao tiếp với DHT11 theo chuẩn 1 chân vi xử lý thực hiện theo 2 bước : gửi tín hiệu muốn đo (Start) tới DHT11, sau đó DHT11 xác nhận lại. Khi đã giao tiếp được với DHT11, cảm biến sẽ gửi lại 5 byte dữ liệu và nhiệt độ đo được.

Arduino UNO R3	Cảm biến nhiệt độ, độ ẩm DHT11
5V	VCC
GND	GND
2	DATA

Hình 2.9. Sơ đồ kết nối chân DHT11

2.5 Cảm biến khí gas MQ-2

2.5.1 Giới thiệu

Cảm biến khí gas MQ-2 sử dụng phần tử SnO₂ có độ dẫn điện thấp hơn trong không khí sạch, khi khí dễ cháy tồn tại, cảm biến có độ dẫn điện cao hơn, nồng độ chất dễ cháy càng cao thì độ dẫn điện của SnO₂ sẽ càng cao và được tương ứng chuyển đổi thành mức tín hiệu điện. Cảm biến xuất ra cả hai dạng tín hiệu là Analog và Digital có thể điều chỉnh mức báo bằng điện trở.

2.5.2 thông số kỹ thuật

- Nguồn hoạt động : 3,3V-5V
- Kích thước PCB : 3cm * 1,6cm
- Led đỏ báo có nguồn vào, led xanh báo có khí gas
- Loại tín hiệu đầu ra : tín hiệu số (digital), tín hiệu tương tự (analog)
- Cấu tạo từ chất bán dẫn SnO₂
- IC so sánh : LM393

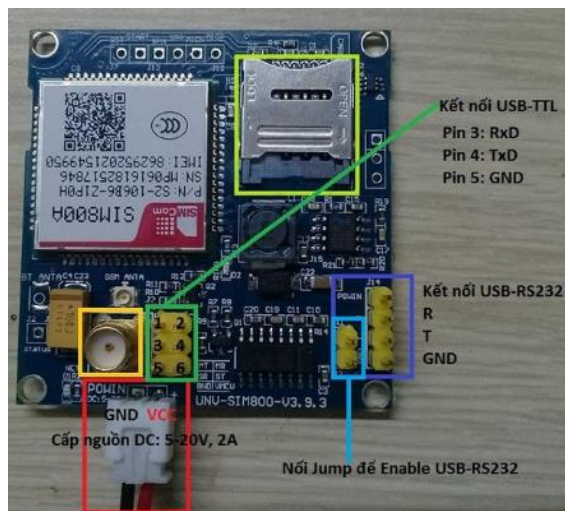
2.6 Module SIM 800A

2.6.1 Giới thiệu

Module SIM800A GSM GPRS Mini là phiên bản thay thế cho module SIM900A đã quá đỗi quen thuộc. Module cải thiện về độ bền, độ ổn định và vẫn có các tính năng cơ bản của một chiếc điện thoại như gọi điện thoại, nhắn tin, truy cập GPRS...

Mạch tích hợp nguồn xung và ic đếm được thiết kế nhỏ gọn nhưng vẫn giữ được các yếu tố cần thiết của thiết kế Module Sim như: Mạch chuyển mức tín hiệu logic sử dụng Mosfet, IC giao tiếp RS232 MAX232, mạch nguồn xung dòng cao, khe sim chuẩn và các đèn led báo hiệu, mạch còn đi kèm với Anten GSM.

2.6.2 Thông số kỹ thuật



Hình 2.10. Module SIM800A

- Sử dụng module GSM GPRS Sim800A.
- Nguồn cấp đầu vào: 5 – 18VDC, lớn hơn 1A.
- Mức tín hiệu giao tiếp: TTL (3.3-5VDC) hoặc RS232.
- Tích hợp chuyển mức tín hiệu TTL Mosfet tốc độ cao.
- Tích hợp IC chuyển mức tín hiệu RS232 MAX232.
- Tích hợp nguồn xung với dòng cao cung cấp cho Sim800A.
- Sử dụng khe Micro Sim.
- Thiết kế mạch nhỏ gọn, bền bỉ, chống nhiễu.

2.6.3 Chức năng các chân

Header 1:

- VCC: Nguồn dương từ 5~18VDC, lớn hơn 1A
- GND: Mass, 0VDC.
- EN: Mặc định nối lên cao, chức năng dùng để khởi động (Enable) hoặc dùng hoạt động (Disable) Module SIM, nếu muốn Module SIM dừng hoạt động bạn có thể nối chân này xuống âm GND (0VDC).
- 232R: Chân nhận tín hiệu RS232 RX.
- 232T: Chân truyền tín hiệu RS232 TX.
- GND: Mass, 0VDC.
- RXD: Chân nhận tín hiệu UART RX TTL 3.3VDC.
- TXD: Chân nhận tín hiệu UART TX TTL 3.3VDC.

Header 2:

- BRXD: Thường không sử dụng, chân nhận tín hiệu, dùng để giao tiếp nạp Firmware cho Sim800, mức tín hiệu 3.3VDC.
- BTXD: Thường không sử dụng, chân truyền tín hiệu, dùng để giao tiếp nạp Firmware cho Sim800, mức tín hiệu 3.3VDC.
- GND: Mass, 0VDC.
- EPN: Ngõ ra loa Speaker âm
- EPP: Ngõ ra loa Speaker dương.
- MICP: Ngõ vào Micro dương.
- MICN: Ngõ vào Micro âm.

2.6.4 Tập lệnh AT của module SIM 800A

❖ Cách tập lệnh chung :

- Lệnh : AT<CR><LF>

Mô tả : Kiểm tra đáp ứng của module SIM 800A, nếu trả về OK thì module hoạt động

- Lệnh : ATE[x]<CR><LF>

Mô tả : Chế độ echo là chế độ phản hồi dữ liệu truyền đến của module SIM 800A, x = 1 bật chế độ echo , x = 0 tắt chế độ echo

- Lệnh : AT + IPR = [band rate]<CR><LF>

Mô tả : cài đặt tốc độ giao tiếp dữ liệu với module SIM 800A chỉ cài được tốc độ baud rate : 0 (auto), 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200.

- Lệnh : AT&W<CR><LF>

Mô tả : lưu lại các lệnh cài đặt

❖ **Các lệnh điều khiển cuộc gọi :**

- Lệnh: AT+CLIP=1<CR><LF>

Mô tả : Hiện thị thông tin cuộc gọi đến

- Lệnh: ATD[Số_điện_thoại];<CR><LF>

Mô tả : Lệnh thực hiện cuộc gọi

- Lệnh: ATH<CR><LF>

Mô tả : Lệnh thực hiện kết thúc cuộc gọi , hoặc cúp máy khi có cuộc gọi đến

- Lệnh: ATA<CR><LF>

Mô tả : Lệnh thực hiện chấp nhận khi có cuộc gọi đến

❖ **Các lệnh điều khiển tin nhắn :**

- Lệnh: AT+CMGF=1<CR><LF>

Mô tả : Lệnh đưa SMS về chế độ Text , phải có lệnh này mới gửi nhận tin nhắn dạng Text

- Lệnh: AT+CMGS="Số_điện_thoại"<CR><LF>

Mô tả : Lệnh gửi tin nhắn

- Lệnh: AT+CMGR=x<CR><LF> (x là địa chỉ tin nhắn cần đọc)

Mô tả : Đọc một nhắn vừa gửi đến, lệnh được trả về nội dung tin nhắn, thông tin người gửi, thời gian gửi

- Lệnh: AT+CMGDA="DEL ALL"<CR><LF>

Mô tả : Xóa toàn bộ tin nhắn trong các hộp thư

- Lệnh: AT+CNMI=2,2<CR><LF>

Mô tả : Hiện thị nội dung tin nhắn ngay khi có tin nhắn đến

2.7 Mạch giảm áp DC LM2596

2.7.1 Giới thiệu

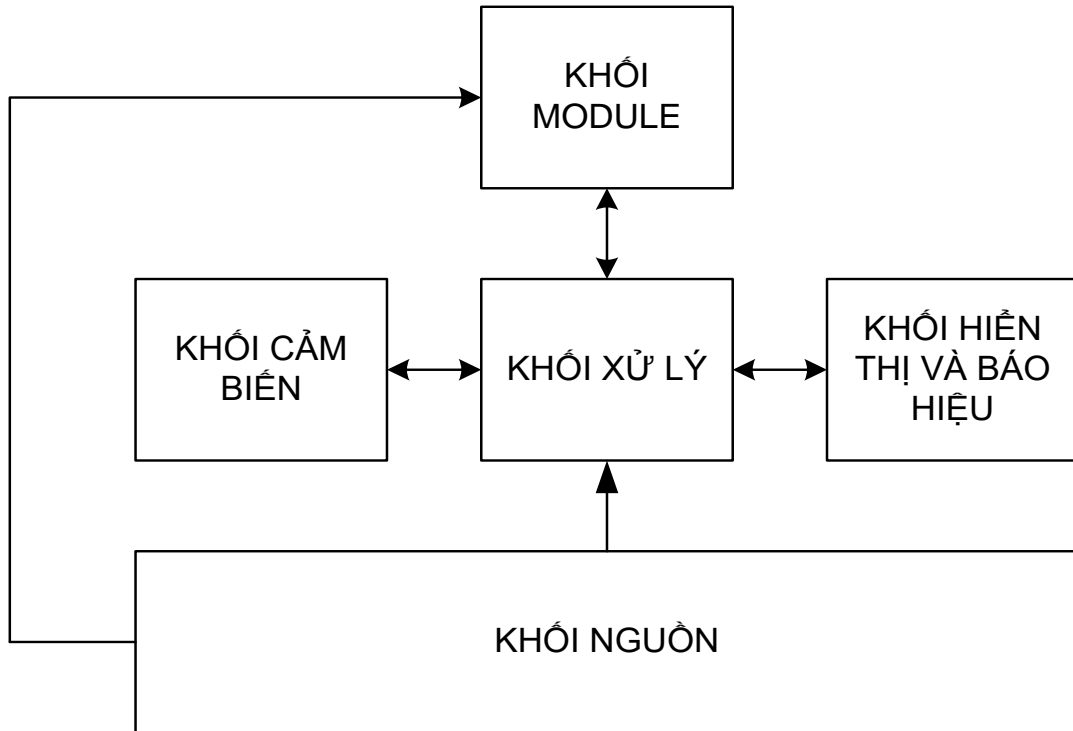
Mạch giảm áp DC LM2596 nhỏ gọn có khả năng giảm áp từ 30V xuống 1.5V mà vẫn đạt hiệu suất cao (92%). Thích hợp cho các ứng dụng chia nguồn, hạ áp, cấp cho các thiết bị như camera, motor, robot,...

2.7.2 Thông số kỹ thuật

- Dùng IC LM2596 với tần số lên đến 150Khz.
- Có nút nhấn chuyển chế độ hiển thị ngõ ra/vào.
- Điện áp đầu vào: Từ 4V đến 30V.
- Điện áp đầu ra: Điều chỉnh được trong khoảng 1.5V đến 29V.
- Dòng ngõ ra tối đa là 3A.
- Công suất: 15W
- Kích thước: 66 x 35mm.

CHƯƠNG 3: CƠ SỞ THỰC HIỆN

3.1 Thiết kế sơ đồ khối của hệ thống



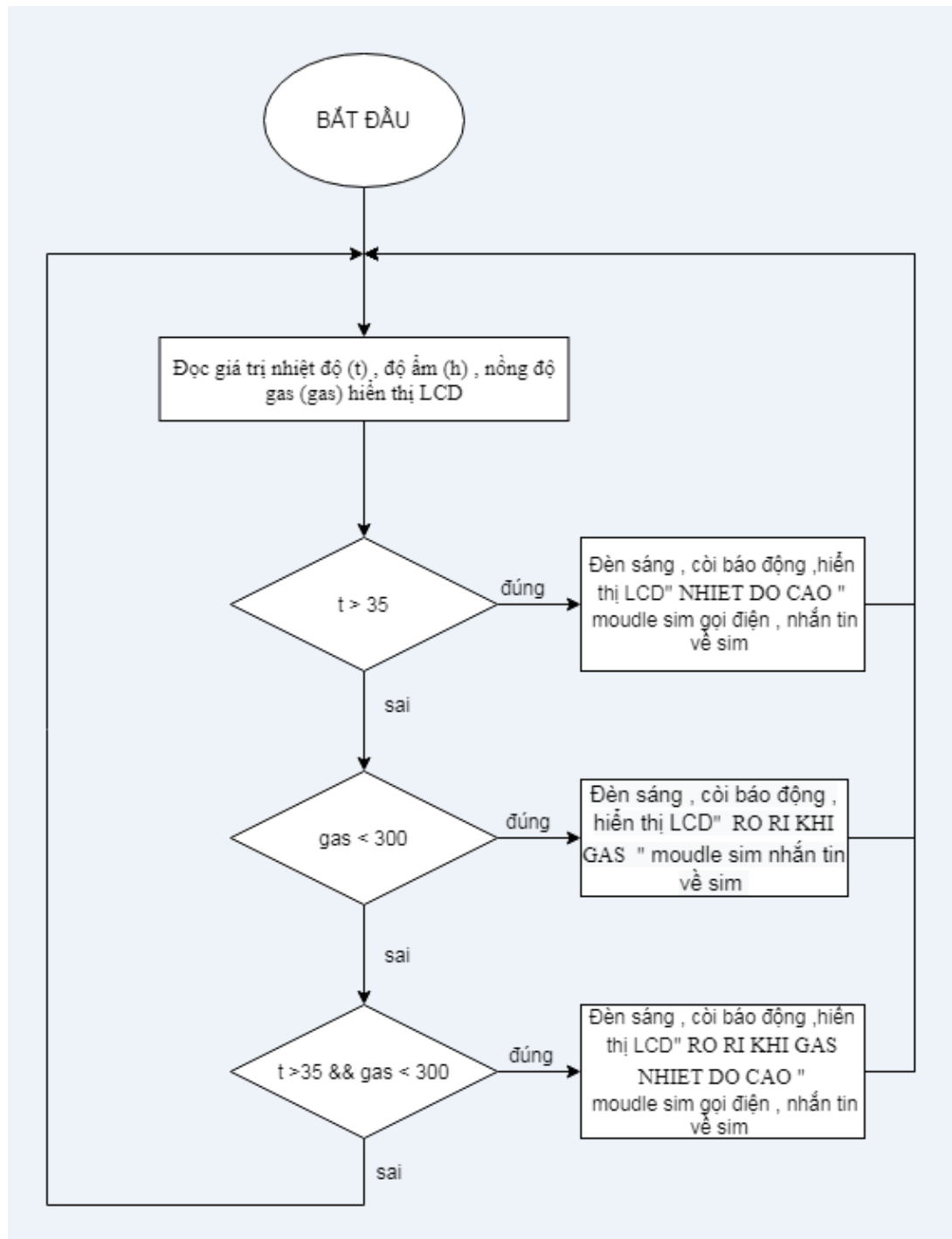
Hình 3.1. Sơ đồ khối của mạch

❖ Chức năng từng khối :

- **Khối hiển thị và báo hiệu:** LCD 16x2 dùng để hiển thị các thông số hoạt động của mạch như: hiển thị thông số nhiệt độ độ ẩm. Loa báo và led sẽ sáng lên mỗi khi nhận được báo động từ khối xử lý.
- **Khối cảm biến:** Bao gồm cảm biến độ ẩm nhiệt độ không khí DHT11, Nhiệm vụ là thu thập các thông số của môi trường và truyền lại cho khối điều khiển và hiển thị
- **Khối Module:** Có chức năng thực thi việc gửi dữ liệu đến thiết bị của người tiêu dùng.
- **Khối Xử lý:** Arduino đóng vai trò là vi điều khiển trung tâm, điều khiển toàn bộ hoạt động của mạch. Dùng để đọc dữ liệu từ cảm biến độ ẩm và nhiệt độ, hiển thị LCD.

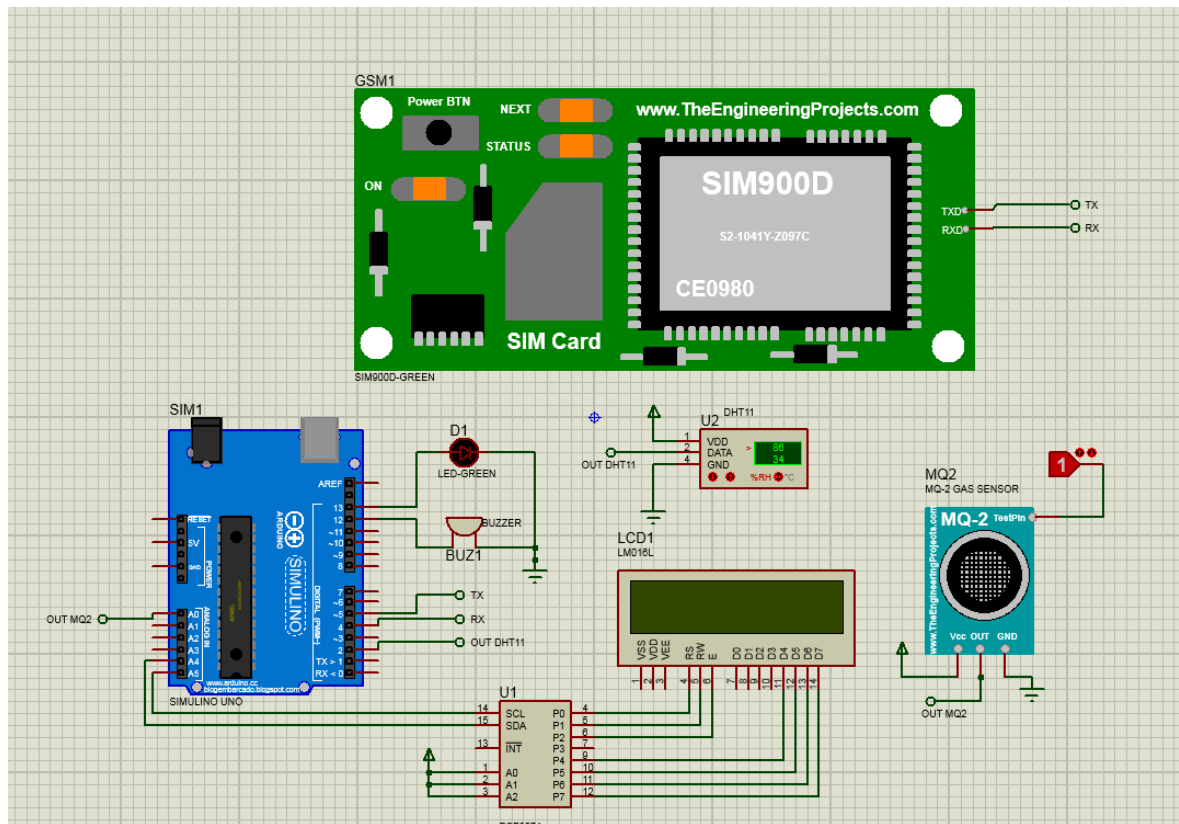
- Khối nguồn: Có chức năng cấp nguồn 12V cho vi điều khiển và khối module (sim800A) và nguồn 5V từ khối xử lý sẽ cấp cho khối cảm biến và khối hiển thị.

3.2 Thiết kế lưu đồ giải thuật



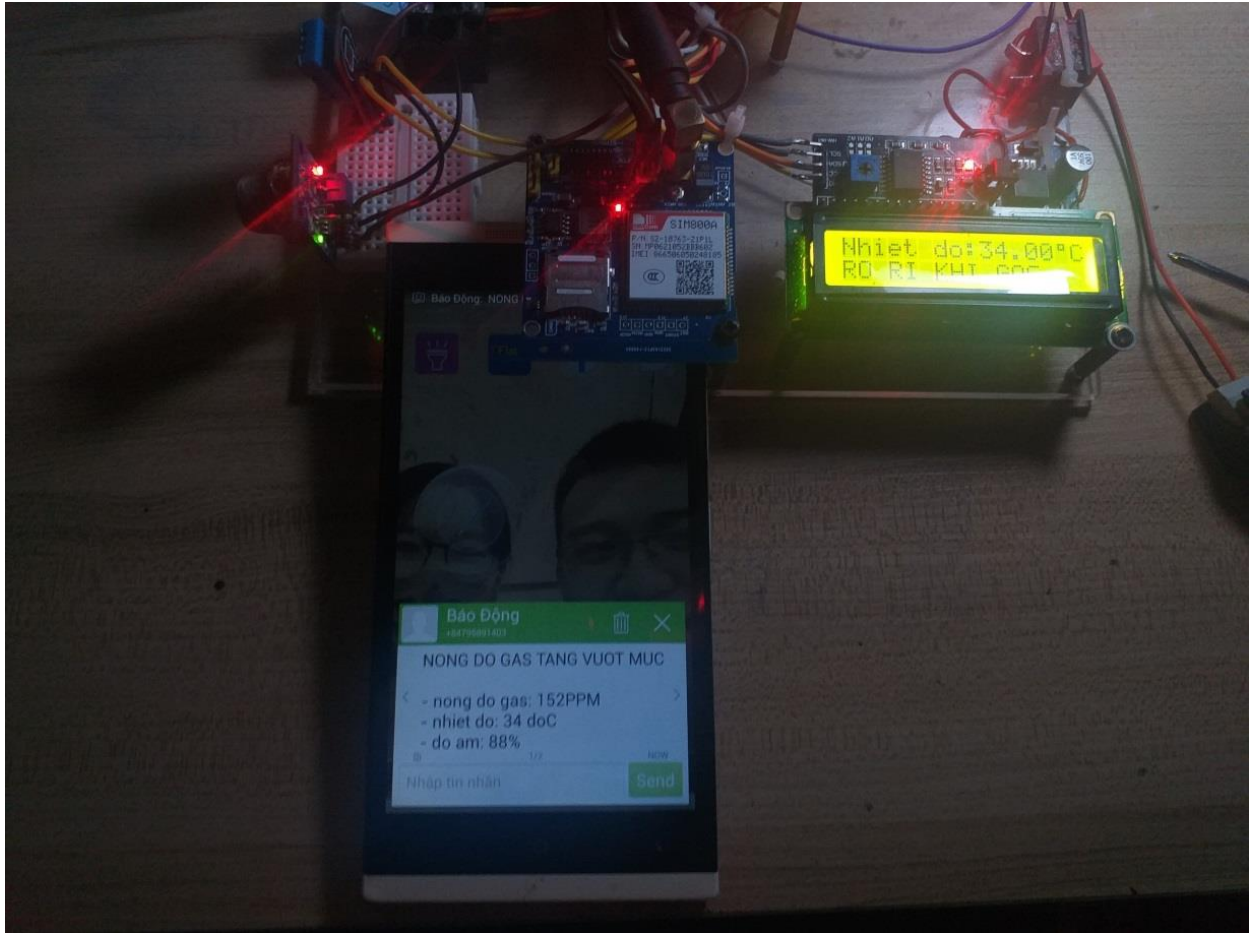
Hình 3.2. Lưu đồ giải thuật

3.3 Thiết kế sơ đồ kết nối của hệ thống



CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM

4.1 Lượng khí gas vượt ngưỡng cho phép

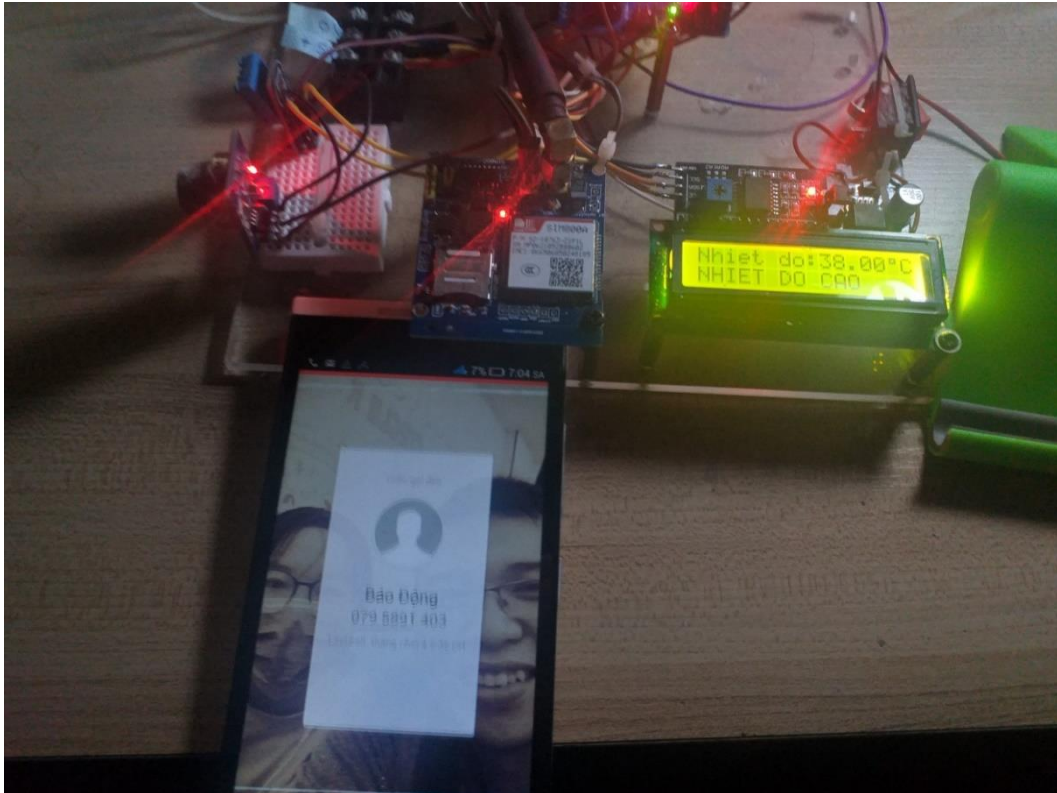


Hình 4.1. Khí gas vượt ngưỡng cho phép

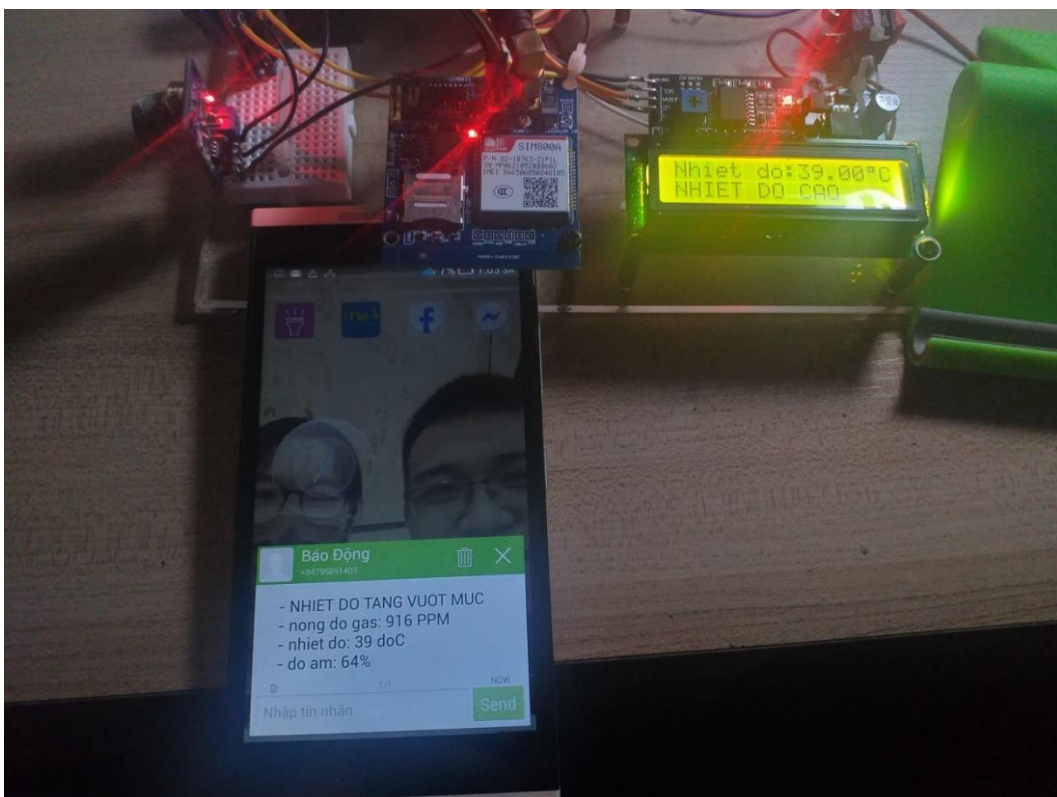
Khí gas vượt ngưỡng cho phép thì đèn sáng , cài báo động , hiển thị cảnh báo lên LCD kèm theo đó sẽ nhắn tin về điện thoại

4.2 Nhiệt độ vượt ngưỡng cho phép

Nhiệt độ vượt ngưỡng cho phép thì đèn sáng , cài báo động , hiển thị cảnh báo lên LCD kèm theo đó sẽ gọi , nhắn tin về điện thoại



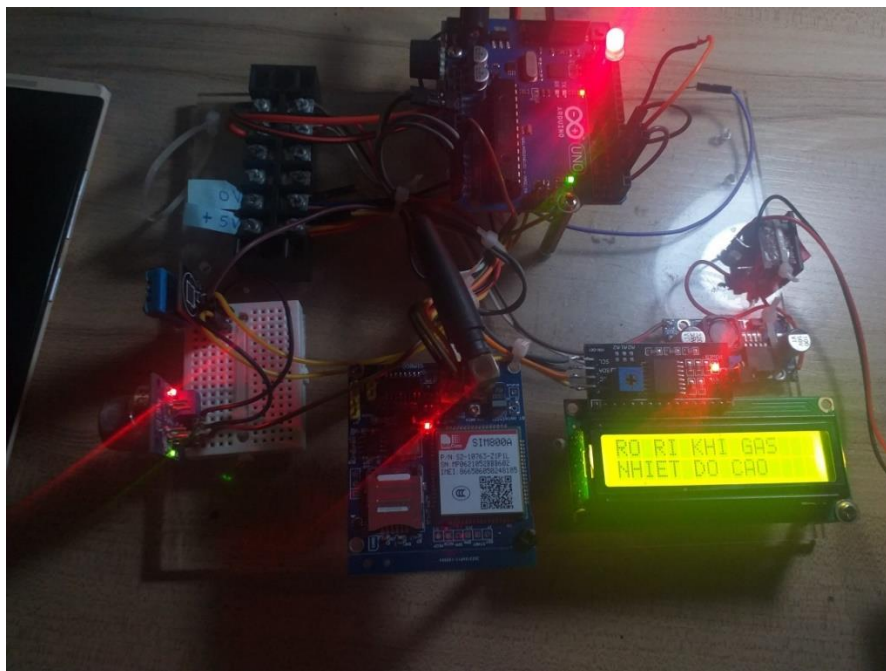
Hình 4.2. Nhiệt độ vượt ngưỡng cho phép



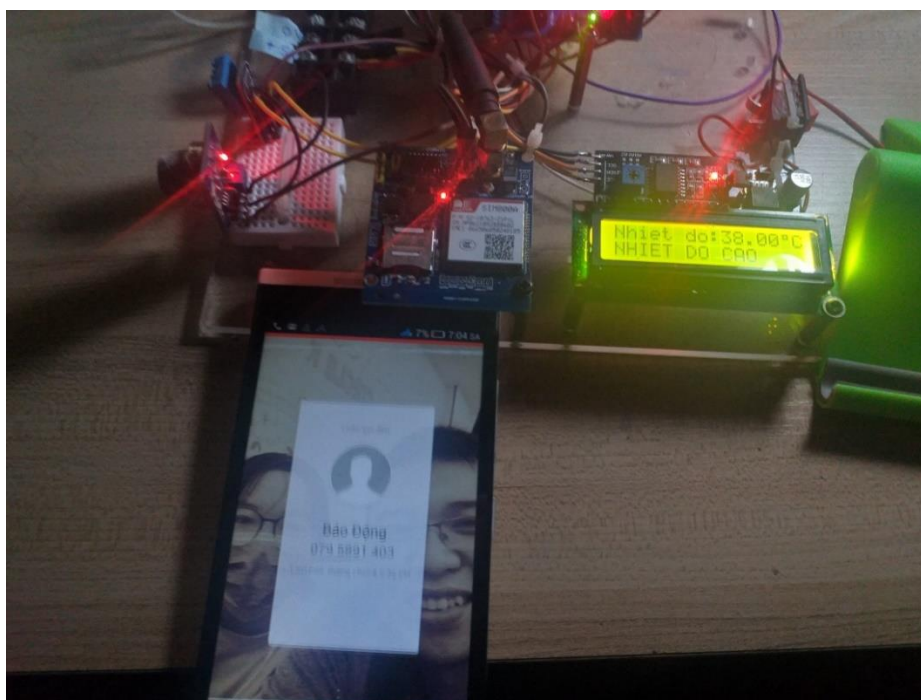
Hình 4.3. Nhiệt độ vượt ngưỡng cho phép

4.3 Nhiệt độ và khí gas cùng vượt ngưỡng cho phép

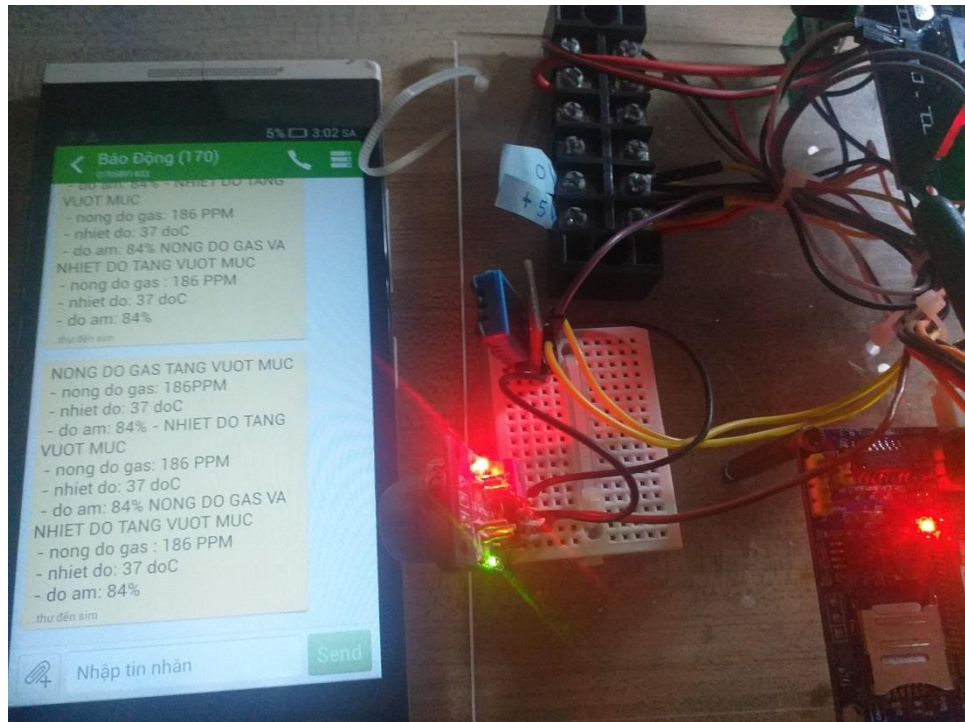
Nhiệt độ và khí gas vượt ngưỡng cho phép thì đèn sáng , cài báo động , hiển thị cảnh báo lên LCD kèm theo đó sẽ gọi , nhắn tin về điện thoại



Hình 4.4. Nhiệt độ và khí gas vượt ngưỡng cho phép



Hình 4.5. Nhiệt độ và khí gas vượt ngưỡng cho phép



Hình 4.6. Nhiệt độ và khí gas vượt ngưỡng cho phép

CHƯƠNG 5: KẾT LUẬN VÀ ĐỊNH HƯỚNG ĐỀ TÀI

5.1 Kết quả đạt được

Sau thời gian học tập, tìm hiểu dưới sự hướng dẫn của GVHD: Ths. Trần Hoàn, đồ án học phần 2 đã hoàn thành đúng thời hạn quy định, giải quyết các yêu cầu đặt ra ban đầu:

- Tìm hiểu về Arduino
- Tìm hiểu cảm biến nhiệt độ DHT11, cảm biến khí gas MQ2
- Tìm hiểu về module sim 800A
- Sản phẩm thực tế

5.2 Hạn chế

- Nhìn chung thì mô hình của em còn nhiều sai sót nhiệt độ, độ ẩm có sự chênh lệch.
- Nhận tín hiệu để gửi tin nhắn về điện thoại còn lâu.
- Mô phỏng đáp ứng chậm
- Mô hình rườm rà, đầu nối không đẹp

5.3 Hướng phát triển của đề tài

- Phát triển thêm hệ thống phun nước tự động giúp chữa cháy hiệu quả hơn.
- Cải thiện thêm diện tích cảm biến rộng hơn.
- Phát triển có thể gửi qua Wifi hoặc bluetooth.

PHỤ LỤC

Code chương trình :

```
#include <DHT.h>

#include <SoftwareSerial.h>

#include <SoftwareSerial.h>

#include <LiquidCrystal_I2C.h>

SoftwareSerial SIM800(4,5);

LiquidCrystal_I2C lcd(0x27,16,2);

int MQ2 = A0;

int DHTPIN = 2;

int DHTTYPE = DHT11;

DHT dht(DHTPIN, DHTTYPE);

int led= 13;

int coi= 12;

byte degree[8] = {

    0B01110,

    0B01010,

    0B01110,

    0B00000,

    0B00000,

    0B00000,

    0B00000,

    0B00000
```

```
};  
  
void setup()  
{  
  pinMode(coi, OUTPUT);  
  pinMode(led, OUTPUT);  
  pinMode(MQ2, INPUT);  
  pinMode(DHTPIN, INPUT);  
  Serial.begin(9600);  
  SIM800.begin(9600);  
  lcd.init();  
  lcd.backlight();  
  lcd.createChar(1, degree);  
  dht.begin();  
}  
  
void loop()  
{  
  int gas = analogRead(A0);  
  float nhietdo = dht.readTemperature();  
  float doam = dht.readHumidity();  
  lcd.setCursor(0,0);  
  lcd.print("Nhiet do: ");  
  lcd.setCursor(0,1);  
  lcd.print("Do am: ");  
  lcd.setCursor(9,0);
```

```
lcd.print(round(nhietdo));  
  
lcd.print("");  
  
lcd.write(1);  
  
lcd.print("C");  
  
lcd.setCursor(6,1);  
  
lcd.print(round(doam));  
  
lcd.print("%    ");  
  
Serial.print("\nnong do gas: ");  
  
Serial.println(gas);  
  
Serial.print("nhiet do: ");  
  
Serial.println(nhietdo);  
  
Serial.print("doam: ");  
  
Serial.println(doam);  
  
delay(200);  
  
if (nhietdo>35)  
{  
  
digitalWrite(led, HIGH);  
  
digitalWrite(coi, HIGH);  
  
lcd.setCursor(0,1);  
  
lcd.print("NHIET DO CAO");  
  
call_on();  
  
delay(2000);  
  
send_sms(gas,nhietdo,doam);  
  
delay(1000);
```

```
}  
  
else  
  
{  
  
digitalWrite(led, LOW);  
digitalWrite(coi, LOW);  
  
}  
  
if (gas < 300 )  
{  
  
digitalWrite(led, HIGH);  
digitalWrite(coi, HIGH);  
  
lcd.setCursor(0,1);  
  
lcd.print("RO RI KHI GAS");  
  
send_sms(gas,nhietdo,doam);  
  
delay(2000);  
  
}  
  
else  
  
{  
  
digitalWrite(led, LOW);  
digitalWrite(coi, LOW);  
  
}  
  
if (gas < 300&&nhietdo>35 )  
{  
  
digitalWrite(led, HIGH);  
  
digitalWrite(coi, HIGH);
```

```
lcd.clear();

lcd.setCursor(0,0);

lcd.print("RO RI KHI GAS");

lcd.setCursor(0,1);

lcd.print("NHIET DO CAO");

call_on();

delay(2000);

send_sms(gas,nhietdo,doam);

delay(1000);

}

else

{

digitalWrite(led, LOW);

digitalWrite(coi, LOW);

}

}

void call_on()

{

SIM800.println("ATD+84929091324;");// goi dien

delay(20000); // thuc hien cuoc goi trong 20s

SIM800.println("ATH");// ket thuc cuoc goi

}
```

```
void send_sms(int gas,int nhietdo, int doam)

{

SIM800.println("AT+CMGF=1");

delay(1000);

SIM800.println("AT+CMGS=\"+84929091324\"\\r"); //sdt nhan tin nhan \\r kiem
tra ki tu cuoi cung

delay(1000);

if(gas<300)

{

SIM800.print(" NONG DO GAS TANG VUOT MUC ");

SIM800.print("\\n - nong do gas: ");

SIM800.print(gas);

SIM800.print("PPM");

SIM800.print("\\n - nhiet do: ");

SIM800.print(nhietdo);

SIM800.print(" doC");

SIM800.print("\\n - do am: ");

SIM800.print(doam);

SIM800.print("%");

delay(100);

}

if(nhietdo>35)

{

SIM800.print(" - NHIET DO TANG VUOT MUC");
```

```
SIM800.print("\n - nong do gas: ");  
  
SIM800.print(gas);  
  
SIM800.print(" PPM");  
  
SIM800.print("\n - nhiet do: ");  
  
SIM800.print(nhietdo);  
  
SIM800.print(" doC");  
  
SIM800.print("\n - do am: ");  
  
SIM800.print(doam);  
  
SIM800.print("%");  
  
delay(100);  
  
}  
  
if(gas <300 && nhietdo>35)  
{  
  
SIM800.print(" NONG DO GAS VA NHIET DO TANG VUOT MUC ");  
  
SIM800.print("\n - nong do gas : ");  
  
SIM800.print(gas);  
  
SIM800.print(" PPM");  
  
SIM800.print("\n - nhiet do: ");  
  
SIM800.print(nhietdo);  
  
SIM800.print(" doC");  
  
SIM800.print("\n - do am: ");  
  
SIM800.print(doam);  
  
SIM800.print("%");  
  
delay(100);
```



```
}  
SIM800.println((char)26); // send sms delay(1000);  
}
```

TÀI LIỆU THAM KHẢO

- [1] <http://arduino.vn/bai-viet/899-huong-dan-su-dung-cam-bien-khi-gas-mq2-voi-arduino>
- [2] <http://arduino.vn/bai-viet/91-doc-nhiệt-do-do-am-va-xuat-ra-man-hinh-lcd>
- [3] <http://arduino.vn/bai-viet/531-dieu-khien-lcd-bang-arduino-uno>
- [4] https://mlab.vn/index.php?_route_=50176-huong-dan-lap-trinh-module-sim800a-va-arduino.html