

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  
**KỸ THUẬT ROBOT VÀ TRÍ TUỆ NHÂN TẠO K62**

-----



**BÁO CÁO BÀI TẬP LỚN**  
**THIẾT KẾ MẠCH ĐO LƯỜNG - CẢM BIẾN**  
**NĂM 2022-2023**

**Sinh viên thực hiện: (Nhóm 9 – Đề 9)**

<b>Họ Và Tên</b>	<b>Mã Sinh Viên</b>
<b>1. Nguyễn Quang Huy</b>	<b>213300715</b>
<b>2. Trần Quốc Huy</b>	<b>213304021</b>
<b>3. Phạm Phúc Hướng</b>	<b>213304367</b>
<b>4. Nguyễn Hữu Khánh</b>	<b>213304493</b>

**Khoa: Điện - Điện Tử**

**Lớp: Kỹ Thuật Robot & Trí Tuệ Nhân Tạo**

**Giảng viên: Thạc sĩ VÕ QUANG SƠN**

**HÀ NỘI\_2023**

## LỜI NÓI ĐẦU

Thực hiện bài tập lớn , nhóm thực hiện đã nhận được nhiều sự quan tâm và tạo điều kiện của Ban Giám hiệu nhà trường, cán bộ văn phòng khoa Điện -Điện tử , trường Đại học Giao thông vận tải. Nhóm thực hiện xin gửi lời cảm ơn sâu sắc đến sự giúp đỡ của Quý Thầy Cô.

Đặc biệt, nhóm thực hiện xin bày tỏ lòng biết ơn sâu sắc và chân thành nhất đến Thầy **Võ Quang Sơn** – Giảng viên trực tiếp hướng dẫn, chỉ bảo tận tình cùng những định hướng và tạo mọi điều kiện thuận lợi nhất để nhóm hoàn thành bài tập lớn này. Giảng viên hướng dẫn đã luôn theo sát từng giao đoạn và có những hỗ trợ, tư vấn kịp thời để tạo nên sản phẩm cuối cùng của nhóm thực hiện.

Mặc dù, nhóm thực hiện đã có nhiều cố gắng để thực hiện bài tập lớn một cách hoàn chỉnh nhất, song công trình làm bài tập khó tránh khỏi những thiếu sót. Nhóm nghiên cứu xin gửi lời cảm ơn đến Quý Thầy Cô trong Hội đồng nghiệm thu bài tập đã đưa ra những ý kiến vô cùng quý báu giúp nhóm thực hiện khắc phục được những thiếu sót trong công trình và góp phần nâng cao chất lượng đào tạo.

*Chúng em xin chân thành cảm ơn!*

**Hà Nội, tháng 4 năm 2023**

**NHÓM THỰC HIỆN**

### THIẾT KẾ MẠCH ĐO LƯỜNG – CẢM BIẾN.

**Đề 9:**

**Câu 1. Thiết kế Mạch đo Nhiệt độ và cảnh báo nhiệt của Hệ thống ấp trứng gia cầm.**  
Thiết bị duy trì bất 3 bóng đèn sưởi để duy trì nhiệt độ trong khoảng 35<sup>0</sup> C-38<sup>0</sup> C.

Thiết bị sẽ hiển thị Nhiệt độ trên LCD và phát cảnh báo bằng Chuông đồng thời Bật thêm 01 đèn sưởi khi nhiệt độ dưới ngưỡng  $35^{\circ}\text{C}$  hoặc giảm 01 đèn sưởi khi nhiệt độ quá ngưỡng  $38^{\circ}\text{C}$ .

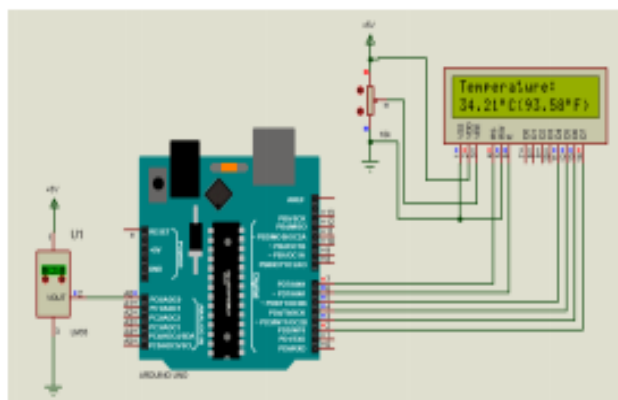
**Thực hiện:**

a. Sử dụng Cảm biến nhiệt độ LM35 để giám sát nhiệt độ. Bắt buộc.

- ✓ Trình bày nguyên lý đo nhiệt độ của cảm biến.
- ✓ Tín hiệu vào, tín hiệu ra.
- ✓ Trình bày các thông số kỹ thuật chính của Cảm biến.

### Trình bày cách thức đọc dữ liệu cảm biến

b. Thiết kế mạch Vi điều khiển dùng Arduino. Về mạch và Mô phỏng dùng Proteus (hoặc phần mềm khác). Tùy chọn.



**Câu 2.** Về lại Sơ đồ mạch Thang đo DC mA : Sanwa, Model YX-360TR.

**Thực hiện:**

- Sử dụng phần mềm vẽ mạch Altium/Proteus/Orcad (hoặc phần mềm khác).

***Yêu cầu trình bày các Nội dung trên trong Báo cáo bài tập Thiết kế mạch Đo lường***

## MỤC LỤC

LỜI NÓI ĐẦU.....	I
ĐỀ BÀI SỐ 1.....	II
MỤC LỤC.....	III
PHỤ LỤC.....	IV
<b>PHẦN 1: THIẾT KẾ MẠCH- SỬ DỤNG CẢM BIẾN NHIỆT ĐỘ .....</b>	<b>1</b>
<b>1.1. TỔNG QUAN VỀ CẢM BIẾN NHIỆT ĐỘ.....</b>	<b>1</b>
1.1.1. Khái niệm cảm biến nhiệt độ LM35.....	1
1.1.2. Sơ đồ chân cảm biến nhiệt độ LM35.....	1
1.1.3. Thông số kỹ thuật của cảm biến LM35.....	2
1.1.4. Nguyên lý hoạt động của cảm biến nhiệt độ LM35 .....	3
1.1.5. Các bước tính toán nhiệt độ bằng cảm biến nhiệt độ LM35.....	3
1.1.6. Công thức chuyển đổi điện áp thành nhiệt độ .....	3
1.1.7. Ứng dụng của cảm biến nhiệt độ LM35.....	3
<b>1.2. THIẾT KẾ VI MẠCH ĐIỀU KHIỂN DÙNG ARDUINO.....</b>	<b>4</b>
1.2.1. Tổng quan các thiết bị sử dụng trong hệ thống.....	4
1.2.1.1. Arduino .....	4
*Tổng quan Arduino.....	4
*Vi điều khiển .....	4
*Các cổng vào ra .....	5
1.2.1.2. Cảm biến LMS35 ( Phần 1.1).....	6
1.2.1.3. Màn hình LCD LM016L.....	7
1.2.1.4. Module giao tiếp I2C.....	8
1.2.1.5. Module Relay.....	9
1.2.1.6. Nút nhấn Button.....	10
1.2.1.7. Đèn LED.....	11
1.2.1.8. Loa buzzer.....	12
1.2.2. Phần mềm lập trình Arduino IDE cho Arduino Uno.....	13-14
1.2.3. Code Arduino .....	15-16
1.2.4. Sơ đồ mạch mô phỏng.....	17
1.2.5. Kết quả.....	17-18
<b>PHẦN 2: VẼ SƠ ĐỒ MẠCH THANG ĐO DC mA; SANWA, MODEL YX-360TR</b>	
2.1. Sơ đồ mạch thang đo DC mA.....	19-20
2.2. Đồng hồ SANWA, model YX-360TR.....	19-20
2.3. Sơ đồ mô phỏng trên Proteus.....	21
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>21</b>

## Phụ Lục

<b>Danh mục hình ảnh.....</b>	
Hình 1: Cảm biến LM35 .....	1
Hình 2: Sơ đồ chân cảm biến LM35.....	1
Hình 3:Cấu trúc phần cứng của Arduino uno r3.....	5
Hình 4: Sơ đồ chân vào / ra.....	6
Hình 5: Vi điều khiển ATmega328.....	6
Hình 6: Sơ đồ chân của ATmega328P.....	7
Hình 7: Màn hình LCD.....	8
Hình 8:Module I2C LCD 16X2..	9
Hình 9:Module relay 5v.....	10
Hình 10:Kí hiệu và cấu tạo button.....	11
Hình 11:Sơ đồ cấu tạo LED.....	12
Hình 12: Quang phổ bức xạ điện tử.....	12
Hình 13: Kí hiệu và cấu tạo buzzer .....	13
Hình 14: Giao diện của phần mềm Arduino Uno R3.....	14
Hình 15:Sơ đồ mạch mô phỏng. ....	17
Hình 16:Mạch mô phỏng nhiệt độ $<35^{\circ}$ .....	17
Hình 17: Mạch mô phỏng nhiệt độ $35^{\circ} < t < 38^{\circ}$ .....	18
Hình 18: Mạch mô phỏng nhiệt độ $>38^{\circ}$ .....	18
Hình 19:Sơ đồ mạch thang đo DC ma.....	19
Hình 20:Đồng hồ SANWA YX 360TR.....	20
Hình 21:Sơ đồ mạch thang đo trên Proteus.....	21
<b>Danh mục bảng.....</b>	
Bảng 1: Các chân của màn hình LCD.....	8
Bảng 2: Các ICON chức năng.....	15

# PHẦN 1: THIẾT KẾ MẠCH- SỬ DỤNG CẢM BIẾN NHIỆT ĐỘ

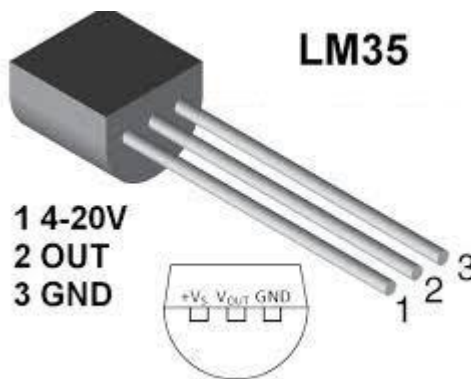
## 1.1.TỔNG QUAN VỀ CẢM BIẾN NHIỆT ĐỘ

### 1.1.1.Khái niệm cảm biến nhiệt độ LMS35

LM35 là một cảm biến nhiệt độ tương tự, điện áp ở đầu ra của cảm biến tỷ lệ với nhiệt độ tức thời và có thể dễ dàng được xử lý để có được giá trị nhiệt độ bằng °C.

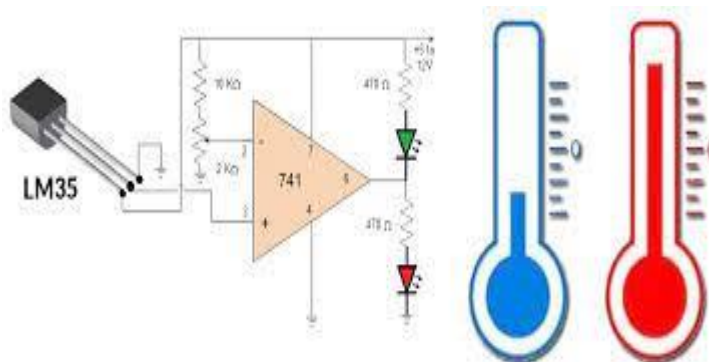
Ưu điểm của LM35 so với cặp nhiệt điện là nó không yêu cầu bất kỳ hiệu chuẩn bên ngoài nào. Lớp vỏ cũng bảo vệ nó khỏi bị quá nhiệt. Chi phí thấp và độ chính xác cao đã khiến cho loại cảm biến này trở thành một lựa chọn đối với những người yêu thích chế tạo mạch điện tử, người làm mạch tự chế và các bạn sinh viên.

Vì có nhiều ưu điểm nêu trên nên cảm biến nhiệt độ LM35 đã được sử dụng trong nhiều sản phẩm đơn giản, giá thành thấp. Đã hơn 15 năm kể từ lần ra mắt đầu tiên nhưng cảm biến này vẫn tồn tại và được sử dụng trong nhiều sản phẩm và ứng dụng đã cho thấy giá trị của loại cảm biến này.



Hình 1: Cảm biến LM35

### 1.1.2: Sơ đồ chân của cảm biến nhiệt độ LM35



Hình 2: Sơ đồ chân cảm biến LM35

Số chân	Tên chân	Chức năng
1	Vcc hay +Vs	Chân cấp nguồn với điện áp từ 4V đến 30V
2	Vout	Chân lấy điện áp ra, điện áp ở chân này thay đổi 10mV/°C
3	GND	Chân nối đất

### 1.1.3: Thông số kỹ thuật của cảm biến LM35

- ✓ Hiệu chuẩn trực tiếp theo °C
- ✓ Điện áp hoạt động: 4-30VDC
- ✓ Dòng điện tiêu thụ: khoảng 60uA
- ✓ Nhiệt độ thay đổi tuyến tính: 10mV/°C
- ✓ Khoảng nhiệt độ đo được: -55°C đến 150°C
- ✓ Điện áp thay đổi tuyến tính theo nhiệt độ: 10mV/°C
- ✓ Độ tự gia nhiệt thấp, 0,08°C trong không khí tĩnh
- ✓ Sai số: 0,25°C
- ✓ Trở kháng ngõ ra nhỏ, 0,2Ω với dòng tải 1mA
- ✓ Kiểu chân: TO92
- ✓ Kích thước: 4.3 × 4.3mm
- ✓ Điện áp hoạt động: 4~20VDC

### 1.1.4: Nguyên lý hoạt động của cảm biến nhiệt độ LM35

Cảm biến LM35 hoạt động bằng cách cho ra một giá trị điện áp nhất định tại chân V<sub>OUT</sub> (chân giữa) ứng với mỗi mức nhiệt độ. Như vậy, bằng cách đưa vào chân bên trái của cảm biến LM35 điện áp 5V, chân phải nối đất, đo hiệu điện thế ở chân giữa, bạn sẽ có được nhiệt độ (0-100°C) tương ứng với điện áp đo được.

Vì điện áp ngõ ra của cảm biến tương đối nhỏ nên thông thường trong các mạch ứng dụng thực tế, chúng ta thường dùng [Op-Amp](#) để khuếch đại điện áp ngõ ra này.

### 1.1.5: Các bước tính toán nhiệt độ bằng cảm biến nhiệt độ LM35

- ✓ Thiết kế mạch.
- ✓ Cấp nguồn cho cảm biến với điện áp từ 4V đến 30V. Chân GND được nối đất.
- ✓ Kết nối chân  $V_{OUT}$  với đầu vào bộ chuyển đổi tương tự sang số hay vi điều khiển.
- ✓ Lấy mẫu đọc ADC để xác định điện áp đầu ra  $V_{OUT}$ .
- ✓ Chuyển đổi điện áp thành nhiệt độ.

### 1.1.6: Công thức chuyển đổi điện áp thành nhiệt độ

Công thức để chuyển đổi điện áp sang nhiệt độ độ C cho LM35 là:

$$\text{Nhiệt độ đo được (}^{\circ}\text{C)} = \text{Điện áp được đọc bởi bộ ADC} / 10 \text{ mV}$$

Tôi chia cho 10 mV vì độ nhạy của cảm biến LM35 là 10mV.

Làm theo các bước và hướng dẫn ở trên, bạn có thể dễ dàng giao tiếp cảm biến LM35 với bất kỳ bộ vi điều khiển nào có chân chuyển đổi tín hiệu tương tự sang số được tích hợp sẵn. Hầu hết tất cả các bộ vi điều khiển ngày nay đều có bộ ADC tích hợp sẵn.

### 1.1.7: Ứng dụng của cảm biến nhiệt độ LM35

Cảm biến nhiệt độ LM35 phù hợp cho các ứng dụng:

- ✓ Học tập nghiên cứu
- ✓ Đo nhiệt độ của một môi trường cụ thể
- ✓ Giám sát nhiệt độ trong hệ thống HVAC
- ✓ Kiểm tra nhiệt độ pin
- ✓ Cung cấp thông tin về nhiệt độ của một linh kiện điện tử khác



## **1.2. THIẾT KẾ VI MẠCH ĐIỀU KHIỂN DÙNG ARDUINO**

### **1.2.1. Tổng quan các thiết bị sử dụng trong hệ thống**

#### **1.2.1.1. Arduino**

##### **▯ Tổng quan Arduino**

Arduino cơ bản là một mã nguồn mở về điện tử được tạo thành từ phần cứng và phần mềm. Đơn giản hơn, Arduino là thiết bị có thể tương tác với ngoại cảnh thông qua các

cảm biến và hành vi được lập trình sẵn. Với thiết bị này việc lắp ráp và điều khiển các thiết bị điện tử sẽ dễ dàng hơn bao giờ hết.

Hiện tại có rất nhiều loại vi điều khiển và đa số được lập trình bằng ngôn ngữ C/C++ hoặc Assembly nên rất khó khăn cho những người có ít kiến thức sâu về điện tử và lập trình. Nó là trở ngại cho mọi người muốn tạo riêng cho mình một món đồ mang tính công nghệ. Song Arduino đã giải quyết được vấn đề này, Arduino được phát triển nhằm đơn giản hóa việc thiết kế, lắp ráp linh kiện điện tử cũng như lập trình trên vi điều khiển và mọi người có thể tiếp cận dễ dàng hơn với thiết bị điện tử mà không cần nhiều về kiến thức điện tử và thời gian.

Những thế mạnh của Arduino so với các nền tảng vi điều khiển khác:

- Chạy trên đa nền tảng: Việc lập trình Arduino có thể thực hiện trên các hệ điều hành khác nhau như Windows, Mac Os, Linux trên Desktop, Android trên di động.
- Ngôn ngữ lập trình đơn giản dễ hiểu.
- Mã nguồn mở: Arduino được phát triển dựa trên nguồn mở nên phần mềm chạy trên Arduino được chia sẻ dễ dàng và tích hợp vào các nền tảng khác nhau.
- Mở rộng phần cứng: Arduino được thiết kế và sử dụng theo dạng modul nên việc mở rộng phần cứng cũng dễ dàng hơn.
- Đơn giản và nhanh: Rất dễ dàng lắp ráp, lập trình và sử dụng thiết bị.
- Dễ dàng chia sẻ: Mọi người dễ dàng chia sẻ mã nguồn với nhau mà không lo lắng về ngôn ngữ hay hệ điều hành mình đang sử dụng.

Arduino được chọn làm bộ não xử lý của rất nhiều thiết bị từ đơn giản đến phức tạp. Trong số đó có một vài ứng dụng thực sự chứng tỏ khả năng vượt trội của Arduino

do chúng có khả năng thực hiện nhiều nhiệm vụ rất phức tạp.

Arduino được biết đến nhiều nhất là phần cứng của nó, nhưng phải có phần mềm để lập trình phần cứng. Cả phần cứng và phần mềm gọi chung là Arduino.

##### **▯ Cấu trúc chung**

Arduino Uno là một bo mạch vi điều khiển dựa trên chip ATmega168 hoặc ATmega328. 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 3: Cấu trúc phần cứng của Arduino Uno R3*

#### ◦ Thông số kỹ thuật của Arduino Uno

- Khối xử lý trung tâm là vi điều khiển Atmega328 họ 8bit.
- 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).

#### ⊞ Các cổng vào ra

Arduino Uno có 14 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào /ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển ATmega328 (mặc định thì các điện trở này không được kết nối).

Arduino có 6 chân analog(A0 đến A5) cung cấp độ phân giải tín hiệu 10 bit ( 0 đến 2<sup>8</sup>

- 1 ) để đọc giá trị điện áp trong khoảng 0V-5V. Với chân AREF trên board, có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu cấp điện áp 2.5V

vào chân này thì có thể dùng các chân analog để đo điện áp trong khoảng từ 0V đến 2.5V với độ phân giải vẫn là 10bit.

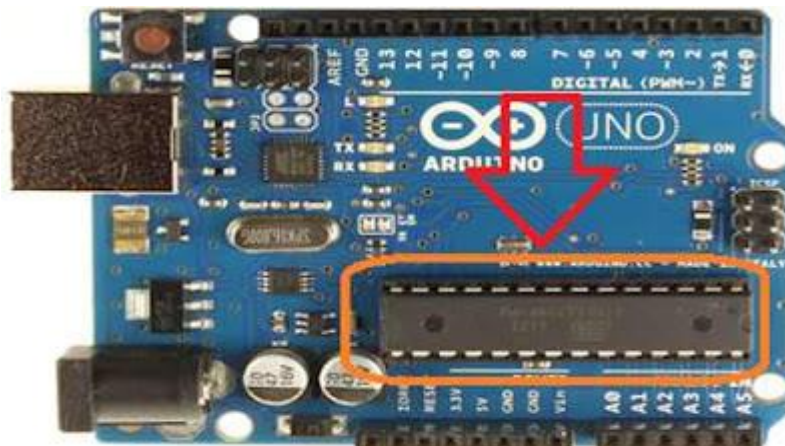
Đặc biệt, Arduino Uno có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.



Hình 4: Sơ đồ chân vào/ra

### ⌘ Vi điều khiển

Arduino Uno R3 có thể sử dụng 3 vi điều khiển họ 8bit AVR là ATmega8, ATmega168, ATmega328. Bộ não này có thể xử lý những tác vụ đơn giản như điều khiển đèn LED nhấp nháy, xử lý tín hiệu cho xe điều khiển từ xa, ... hay những ứng dụng khác. Thiết kế tiêu chuẩn của Arduino Uno R3 sử dụng là ATmega328.



Hình 5: Vi điều khiển ATmega328


Vi điều khiển ATmega328 tiêu chuẩn cung cấp:

• **32KB bộ nhớ Flash:** những đoạn lệnh lập trình sẽ được lưu trữ trong bộ nhớ Flash của vi điều khiển. Thường thì sẽ có khoảng vài KB trong số này sẽ được dùng cho bootloader.

• **2KB cho SRAM:** giá trị các biến khai báo khi lập trình sẽ lưu ở đây. Khi khai báo càng nhiều biến thì càng cần nhiều bộ nhớ RAM. Tuy vậy, thực sự thì cũng hiếm khi nào bộ nhớ RAM lại trở thành thứ phải bận tâm. Khi mất điện, dữ liệu trên SRAM sẽ bị mất.

• **1KB cho EEPROM:** đây giống như một chiếc ổ cứng mini- nơi có thể đọc và ghi dữ liệu của mình vào đây mà không phải lo bị mất khi mất điện giống như dữ liệu trên SRAM.

*Sơ đồ chân của vi điều khiển ATmega328P:*

ARDUINO PINS	ATMEGA328P PIN DETAILS WITH ARDUINO FUNCTIONS										ARDUINO PINS
Reset	(PCINT14/RESET)	PC6	Pin1		Pin28	PC5	(ADC5/SCL/PCINT13)				Analog Input 5
Digital Pin 0 (RX)	(PCINT16/RXD)	PD0	Pin2		Pin27	PD4	(ADC4/SDA/PCINT12)				Analog Input 4
Digital Pin 1 (RX)	(PCINT17/TXD)	PD1	Pin3		Pin26	PD3	(ADC3/PCINT11)				Analog Input 3
Digital Pin 2	(PCINT18/INT0)	PD2	Pin4		Pin25	PC2	(ADC2/PCINT10)				Analog Input 2
Digital Pin 3 (PWM)	(PCINT19/OC2B/INT1)	PD3	Pin5		Pin24	PC1	(ADC1/PCINT9)				Analog Input 1
Digital Pin 4		PD4	Pin6		Pin23	PC0	(ADC0/PCINT8)				Analog Input 0
Vcc		Vcc	Pin7		Pin22	GND					GND
GND		GND	Pin8		Pin21	AREF					Analog Reference
Crystal	(PCINT6/XTAL1/TOSC1)	PB6	Pin9		Pin20	AVCC					Vcc
Crystal	(PCINT7/XTAL2/TOSC2)	PB7	Pin10		Pin19	PB5	(SCK/PCINT5)				Digital Pin 13
Digital Pin 5 (PWM)	(PCINT21/OC0B/T1)	PD5	Pin11		Pin18	PB4	(MISO/PCINT4)				Digital Pin 12
Digital Pin 6 (PWM)	(PCINT22/OC0A/AIN0)	PD6	Pin12		Pin17	PB3	(MOSI/OC2A/PCINT3)				Digital Pin 11(PWM)
Digital Pin 7	(PCINT23/AIN1)	PD7	Pin13		Pin16	PB2	(SS/OC1B/PCINT2)				Digital Pin 10(PWM)
Digital Pin 8	(PCINT0/CLKO/ICP1)	PB0	Pin14		Pin15	PB1	(OC1A/PCINT1)				Digital Pin 9(PWM)

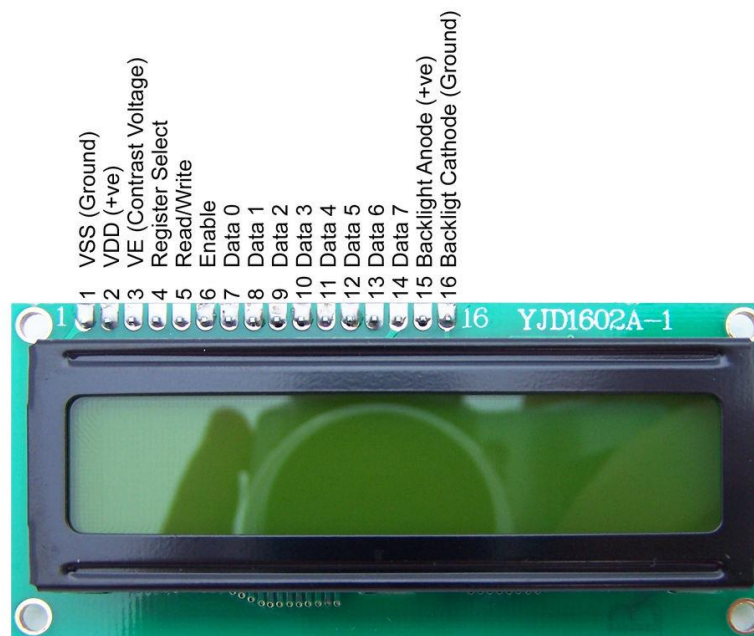
*Hình 6: Sơ đồ chân của ATmega328P*

### 1.2.1.2. Cảm biến nhiệt độ LM35 (Phần 1.1)

### 1.2.1.3. Màn hình LCD LM016L

Chân số 1 – VSS	Chân nối đất cho LCD được nối với GND của mạch điều khiển
Chân số 2 -VDD	Chân cấp nguồn cho LCD, nối với VCC = 5V của mạch điều khiển
Chân số 3 - VEE	Điều chỉnh độ tương phản của LCD
Chân số 4 - RS	Chân chọn thanh ghi, được nối với logic “0” hoặc logic “1”
Chân số 5 – R/W	Chân chọn chế độ đọc/ghi, được nối với logic “0” để ghi hoặc logic “1” để đọc
Chân số 6 - E	Chân cho phép, sau khi cái tín hiệu đã được đặt lên Bus
DB0 – DB7	-Logic “0”: Bus DB0 – DB7 sẽ nối với thanh ghi lệnh IR của LCD (ở chế độ ghi) hoặc nối với bộ đếm địa chỉ của LCD(ở chế độ đọc) -Logic “1”: Bus DB0 – DB7 sẽ nối với thanh ghi dữ liệu DR bên trong LCD
Chân 15 - A	Cực dương Led nền
Chân 16 - K	Cực âm Led nền

*Bảng 1: Các chân của màn hình LCD*



*Hình 7: Màn hình LCD*

#### **1.2.1.4. Module giao tiếp I2C**

Δ LCD có quá nhiều chân gây khó khăn trong quá trình đấu nối và chiếm dụng nhiều chân trên vi điều khiển.

Δ Module I2C ra đời để giải quyết vấn đề này.



Δ Thay vì phải sử dụng 6 chân vi điều khiển để kết nối với LCD 16x2(RS, EN, D7, D6, D5 và D4) thì module I2C chỉ cần tốn 2 chân (SCL, SDA) để kết nối.

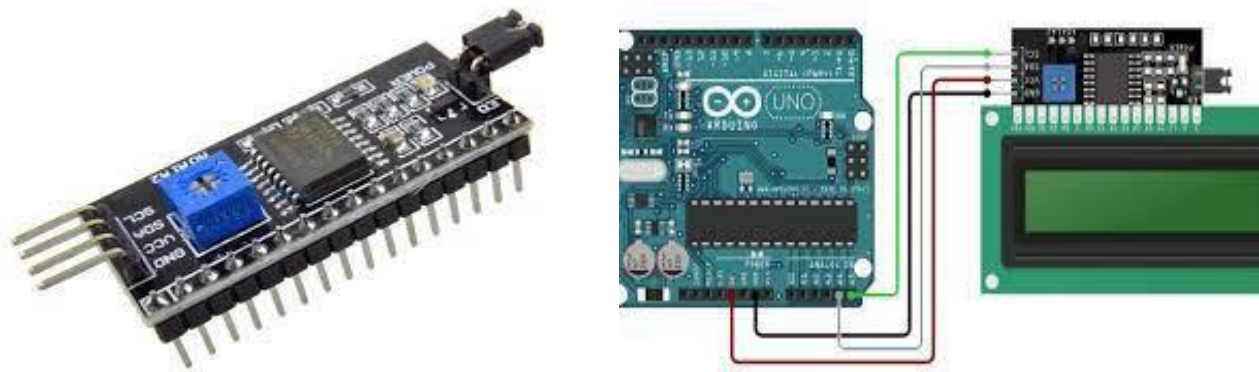
Δ Module I2C hỗ trợ các loại LCD sử dụng driver HD44780(LCD 16x2, LCD 20x4,...) và tương thích với hầu hết các vi điều khiển hiện nay

#### a.Ưu điểm

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

#### b.Thông số kĩ thuật

- Điện áp hoạt động: 2.5-6V DC.
- Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2).
- Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt.
- Tích hợp biến trở xoay điều chỉnh độ tương phản cho LCD



Hình 8: Module I2C LCD 16x2

#### 1.2.1.5.Module Relay

Module Relay 1 kênh 5V gồm 1 rơ le điện áp hoạt động ở mức 5VDC, đầu ra điều khiển hiệu điện tối đa ở mức 250V 10A đối với điện áp xoay chiều AC và 30V với điện áp 1 chiều DC.

Module relay 1 kênh nhỏ gọn chuyên nghiệp, khả năng chống nhiễu tốt và khả năng cách điện tốt. Trong module đã có sẵn mạch kích relay sử dụng IC cách ly quang và transistor giúp cách ly hoàn toàn mạch vi điều khiển với rơ le bảo đảm vi điều khiển hoạt động ổn định.

Có sẵn header rất tiện dụng khi kết nối với vi điều khiển. Có các lỗ bắt vít rất tiện lợi để lắp đặt trong hệ thống mạch.

Mạch điều khiển relay 1 kênh này sử dụng chân kích mức Thấp (0V), khi có tín hiệu 0V vào chân IN thì relay sẽ nhảy qua thường Mở của Relay.  
Ứng dụng với chân relay module khá nhiều bao gồm cả điện DC hay AC.



*Hình 9: Module Relay 5v*

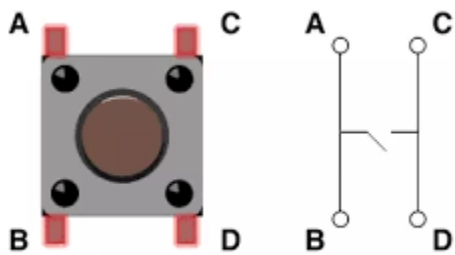
**Thông số kỹ thuật:**

- Kích thước: 53mm (chiều dài) \* 28.3mm (chiều rộng) \* 19.3mm (H)
- Trọng lượng: 18g
- Màu sắc: Đen
- Bốn lỗ để bắt vít cố định có đường kính 3.1mm, dễ dàng lắp đặt trong hệ thống
- Opto cô lập, tốt chống nhiễu
- Có đèn báo đóng ngắt trên Relay.
- Sử dụng điện áp nuôi DC 5V.
- Đầu ra điện thế đóng ngắt tối đa: DC 30V / 10A, AC 250V / 10A

**Cách đấu nối dây:**

- VCC: nối với nguồn 5v
- GND: nối với 0V
- IN: nối với dây tín hiệu điều khiển relay

**1.2.1.6: Nút nhấn button**



*Hình 10: Kí hiệu và cấu tạo button*

Giống với công tắc đóng / mở bạn thấy ở bất cứ đâu, nút nhấn cũng có cơ chế hoạt động giống như vậy. Thay vì chỉ có 2 chân như công tắc, nút nhấn có 4 chân chia làm 2 cặp. Những chân trong cùng một cặp được nối với nhau, những chân khác cặp thì ngược lại. Khi bạn nhấn nút, cả 4 chân của nút nhấn đều được nối với nhau, cho phép dòng điện từ một chân bất kì có thể tới 3 chân còn lại, khi ngừng nhấn, 2 cặp sẽ tách rời, dòng điện sẽ không còn liên thông nữa.

### **1.2.1.7: Đèn LED**

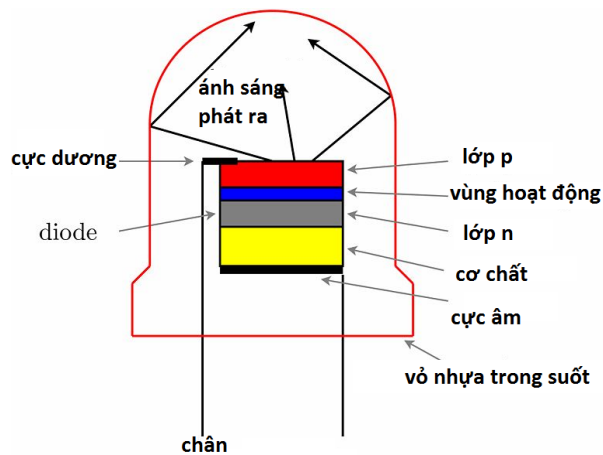
#### **a. Đèn LED là gì?**

LED là từ viết tắt của Light Emitting Diode - Diode phát quang. LED là thiết bị bán dẫn tạo ra ánh sáng. Lúc đầu nó được sử dụng làm đèn báo nhưng sau này được dùng rộng rãi làm đèn chiếu sáng trong nhà, ngoài trời, trang trí

#### **b. Cấu tạo và nguyên lý của LED**

Cấu tạo của đèn led có 1 cực dương (cathode) và một cực âm (anode) ngăn cách nhau bởi một tinh thể vật liệu bán dẫn. Vật liệu bán dẫn được thêm tạp chất để tạo ra các mối nối PN. Toàn bộ được lắp trong một vỏ nhựa có tác dụng như một thấu kính để hướng ánh sáng phát ra ngoài.

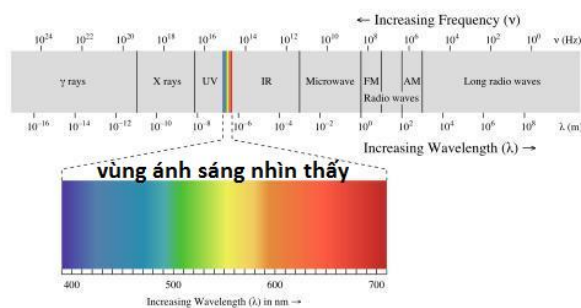




Hình 11: Sơ đồ cấu tạo của led

Khi có điện áp đặt lên các điện cực, dòng điện sẽ đi từ anode (mặt P) sang cathode (mặt N). Khi 1 electron gặp 1 lỗ trống ở chỗ mối nối PN nó bị rơi vào trạng thái năng lượng thấp. Sự khác biệt năng lượng giữa hai trạng thái chính là đặc tính của mối nối PN.

Năng lượng thừa của electron sẽ phát ra thành một photon. Khoảng cách năng lượng càng cao thì độ dài sóng của ánh sáng phát ra càng ngắn.



Hình 12: Quang phổ bức xạ điện từ

Trong dải ánh sáng nhìn thấy thì ánh sáng đỏ có bước sóng dài nhất là 700nm (ít năng lượng nhất) trong khi đó ánh sáng tím có bước sóng ngắn nhất 400nm (nhiều năng lượng nhất)

### c. Các loại LED

Led xuyên lỗ, led SMD, led 2 màu, led RGB, led công suất cao

#### 1.2.1.8: Loa Buzzer



Hình 13: Kí hiệu và cấu tạo buzzer

**Buzzer** là một thiết bị tạo ra tiếng còi hoặc tiếng bíp. Có nhiều loại nhưng cơ bản nhất là buzzer áp điện, là một miếng phẳng của vật liệu áp điện với hai điện cực. Loại buzzer này đòi hỏi phải có các bộ dao động (hoặc vi điều khiển) để điều khiển nó. Nếu bạn sử dụng điện áp một chiều, nó chỉ kêu lách cách. Chúng được sử dụng ở những vị trí cần phát ra âm thanh nhưng không quan tâm đến việc tái tạo âm thanh trung thực, như lò vi sóng, báo cháy và đồ chơi điện tử.

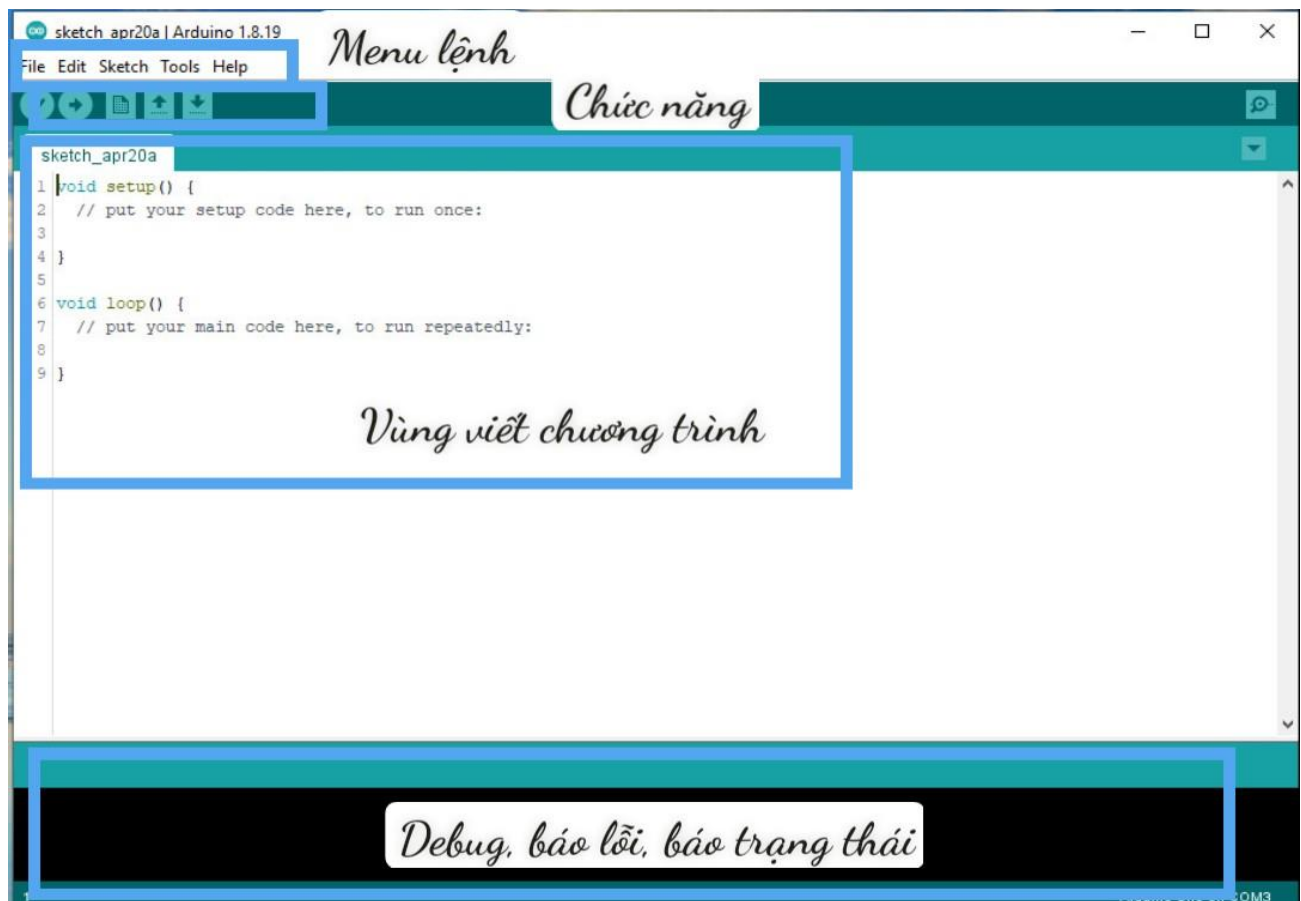
### Thông số kỹ thuật:

- Nguồn : 3.5V - 5.5V
- Dòng điện tiêu thụ: <25mA
- Tần số cộng hưởng: 2300Hz  $\pm$  500Hz
- Biên độ âm thanh: >80 dB
- Nhiệt độ hoạt động:-20 °C đến +70 °C
- Kích thước : Đường kính 12mm, cao 9,7mm

#### 1.2.2. Phần mềm lập trình Arduino IDE cho Arduino Uno

Phần mềm Arduino IDE là trình biên dịch hỗ trợ soạn thảo và nạp code xuống các bản mạch như Arduino Uno R3, Arduino Atmega 2560, Arduino Ethernet, ESP8266...

Arduino ide được tạo ra bởi ngôn ngữ lập trình Java chính là ứng dụng đa nền tảng(cross-platform). Ngôn ngữ lập trình sử dụng cho các chương trình trong Arduino là C hoặc C++. Bản thân Arduino IDE đã được trang bị một thư viện phần mềm thường gọi là “wiring”, từ các chương trình “wiring” gốc đó sẽ giúp thực hiện thao tác lập trình dễ dàng và nhanh hơn. Một chương trình chạy trong Arduino được gọi là một sketch, chương trình được định dạng dưới dạng .ino.



Hình 14: Giao diện của phần mềm Arduino Uno R3

Về cơ bản thì trình biên dịch gồm 4 vùng cơ bản để người lập trình có thể thao tác và xử lý thông tin từ các vùng đó.







**Vùng menu lệnh:** Bao gồm các nút lệnh menu (File, Edit, Sketch, Tools, Help). Trong đó, có 1 số lệnh thường dùng như: open file, save file

Phần Example dùng code có sẵn để tham khảo như: bật tắt led, digital, analog...

Tools --> board: để kết nối với các loại bo mạch, vi điều khiển khác

Tools--> serial port: chọn cổng COM, đó là ột số lệnh thông dụng và còn nhiều lệnh khác nữa....

**Vùng chức năng:**

Icon	Chức năng
	Biên dịch chương trình đang soạn thảo để kiểm tra các lỗi lập trình.
	Biên dịch và upload chương trình đang soạn thảo.
	Mở một trang soạn thảo mới.
	Mở các chương trình đã lưu.
	Lưu chương trình đang soạn.
	Mở cửa sổ Serial Monitor để gửi và nhận dữ liệu giữa máy tính và board Arduino.

*Bảng 2: Các ICON chức năng thường dùng*

**Vùng viết chương trình:** Đây là vùng để ta viết những đoạn mã chương trình. Tên chương trình được hiển thị ngay dưới các Icon, ở đây nó tên là “Blink”. Để ý rằng phía sau tên chương trình có một dấu “\$”. Điều đó có nghĩa là đoạn chương trình chưa được lưu lại.

**Vùng thông báo (debug):** Những thông báo từ IDE sẽ được hiển thị tại đây. Để ý rằng góc dưới cùng bên phải hiển thị loại board Arduino và cổng COM được sử dụng. Luôn chú ý tới mục này bởi nếu chọn sai loại board hoặc cổng COM sẽ không thể upload được chương trình lập trình.

### 1.2.3. Code Arduino

```
LiquidCrystal_I2C lcd(0x27,16,2); // Khai báo LCD
```

- Đặt địa chỉ LCD là 0x27 cho màn hình LCD 16x2.

- 16 là số cột của màn hình.

- 2 là số dòng của màn hình.

**Từ các dữ liệu và kiến thức trên, ta có mã nguồn Arduino:**

```
#include <LiquidCrystal.h> // khai bao thu vien lcd sử dụng i2c
```

```
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);
```

```
const int lampPin = 2;
```

```
const int lampPin1 = 3;
```

```
const int lampPin2 = 4;
```

```
const int temperaturesensorpin = A0;
```

```
float temp = 0;
```

```
float temperature = 0;
```

```
void setup()
```

```

{
  pinMode(lampPin, OUTPUT);
  pinMode(lampPin1, OUTPUT);
  pinMode(lampPin2, OUTPUT);
  lcd.begin(16, 2);      // khai tạo lcd
  lcd.setCursor(0, 0);   // đưa con trỏ tới hàng 0 cột 0
  lcd.print("nhom 9");   // in ra màn hình chữ thành lộn
  lcd.setCursor(0, 1);
  lcd.print("cam bien lm35 ");
  delay(1000);
  lcd.clear();
}

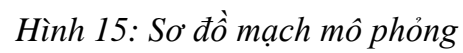
void loop()
{
  temp = analogRead(temperaturesensorpin);
  temperature = (5.0 * temp / 1024.0) * 100.0; // Chuyển đổi giá trị đọc sang độ C
  if(temperature < 35) {
    lcd.setCursor(0, 0);
    lcd.print("temperature");
    lcd.setCursor(0, 1);
    lcd.print(temperature);
    digitalWrite(lampPin, HIGH);
    digitalWrite(lampPin1, HIGH);
    digitalWrite(lampPin2, HIGH);
  }

  else if(temperature >38) {
    lcd.setCursor(0, 0);
    lcd.print("temperature");
    lcd.setCursor(0, 1);
    lcd.print(temperature);
    digitalWrite(lampPin, HIGH);
    digitalWrite(lampPin1, LOW);
    digitalWrite(lampPin2, LOW);
  }

  else {
    lcd.setCursor(0, 0);
    lcd.print("temperature");
    lcd.setCursor(0, 1);
    lcd.print(temperature);
    digitalWrite(lampPin, HIGH);
    digitalWrite(lampPin1, HIGH);
    digitalWrite(lampPin2, LOW);
  }
}

```

#### 1.2.4.Sơ đồ mạch mô phỏng

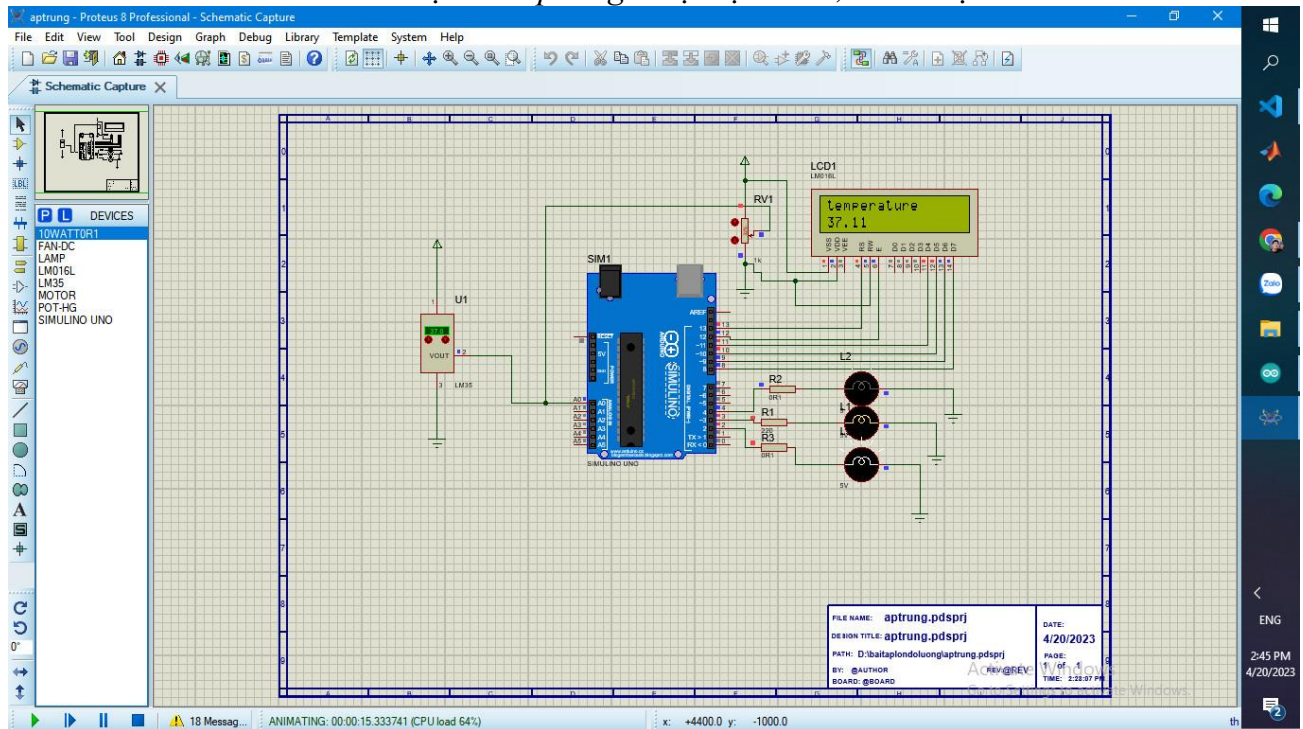


### 1.2.5: Kết quả:

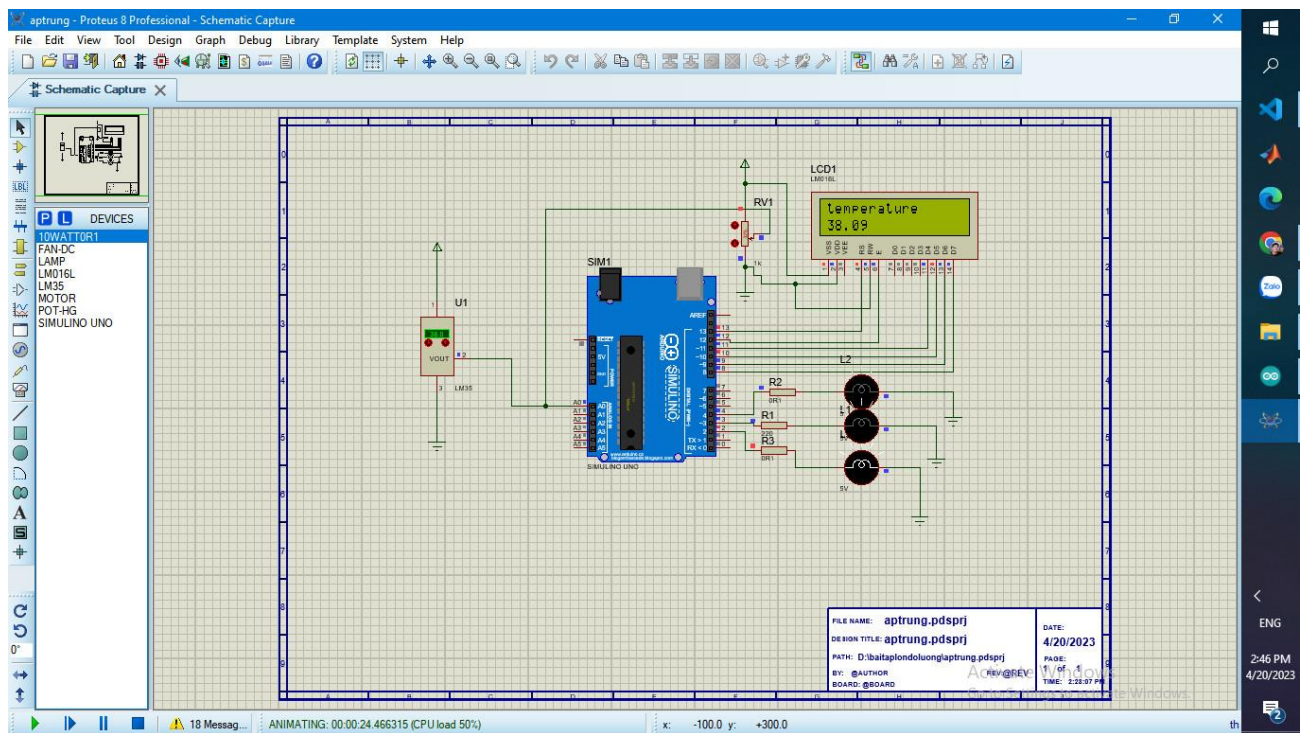




Hình 16: Mạch mô phỏng nhiệt độ  $< 35^\circ$ , 3 đèn bật.



Hình 17: Mạch mô phỏng nhiệt độ từ  $35^\circ < t < 38^\circ$ , đèn 1 tắt

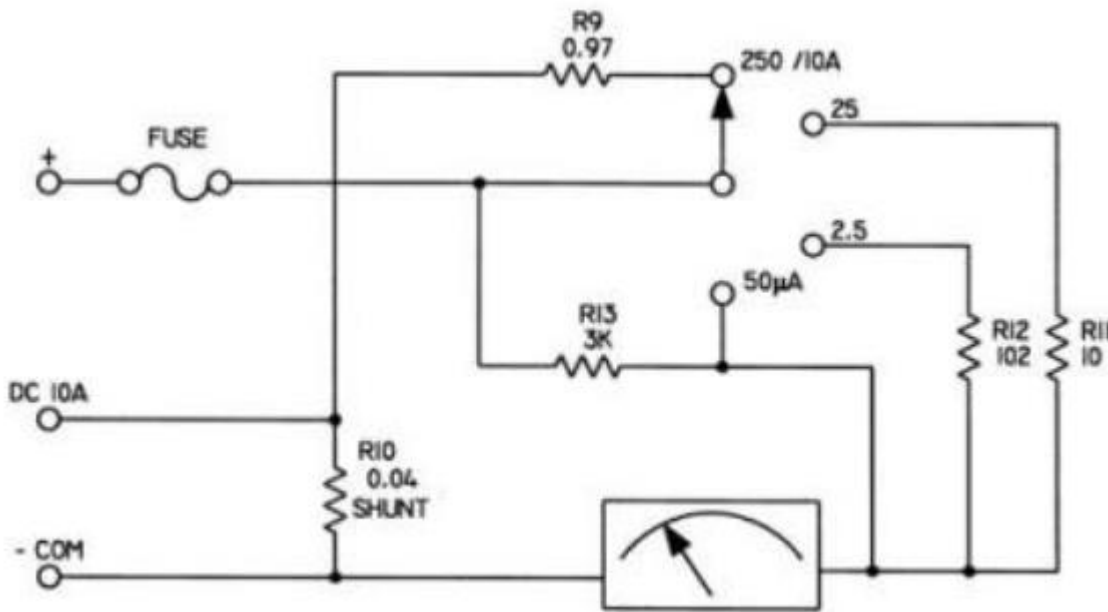


Hình 18: Mạch mô phỏng nhiệt độ  $> 38^\circ$ , đèn 1 và đèn 2 tắt.

## PHẦN 2: VẼ SƠ ĐỒ MẠCH THANG ĐO DC mA; Sanwa, model YX-360TR

### 2.1. Sơ đồ mạch thang đo DC mA

Trên dải 0.01mA, dòng điện được đưa trực tiếp vào đồng hồ và điện áp rơi trên đồng hồ ở độ lệch toàn thang đo là 0,1V. Trên tất cả các dải khác, điện áp rơi trên toàn thang đo đồng hồ là 0,25V. Mạch được lắp thêm một cầu chì 0.5A để tránh quá tải. Sơ đồ mạch đơn giản của mạch đo dòng điện một chiều(DC) bao gồm 5 điện trở R1, R2, R3, R4, điện trở shunt và một cầu chì.



Hình 19: Sơ đồ mạch thang đo DC mA

### 2.2: Đồng hồ Sanwa Model YX – 360TR

Đồng hồ Sanwa Model YX-360TR là một dụng cụ đo lường điện có nhiều chức năng. Các chức năng cơ bản là ampe kế, vôn kế, và ôm kế, ngoài ra có một số đồng hồ còn có thể đo tần số dòng điện, điện dung tụ điện, kiểm tra bóng bán dẫn (transistor)...

#### Tính năng:

- Thiết bị được làm từ chất liệu đàn hồi, chống sốc khi rơi. Vật liệu cao cấp, độ bền cao

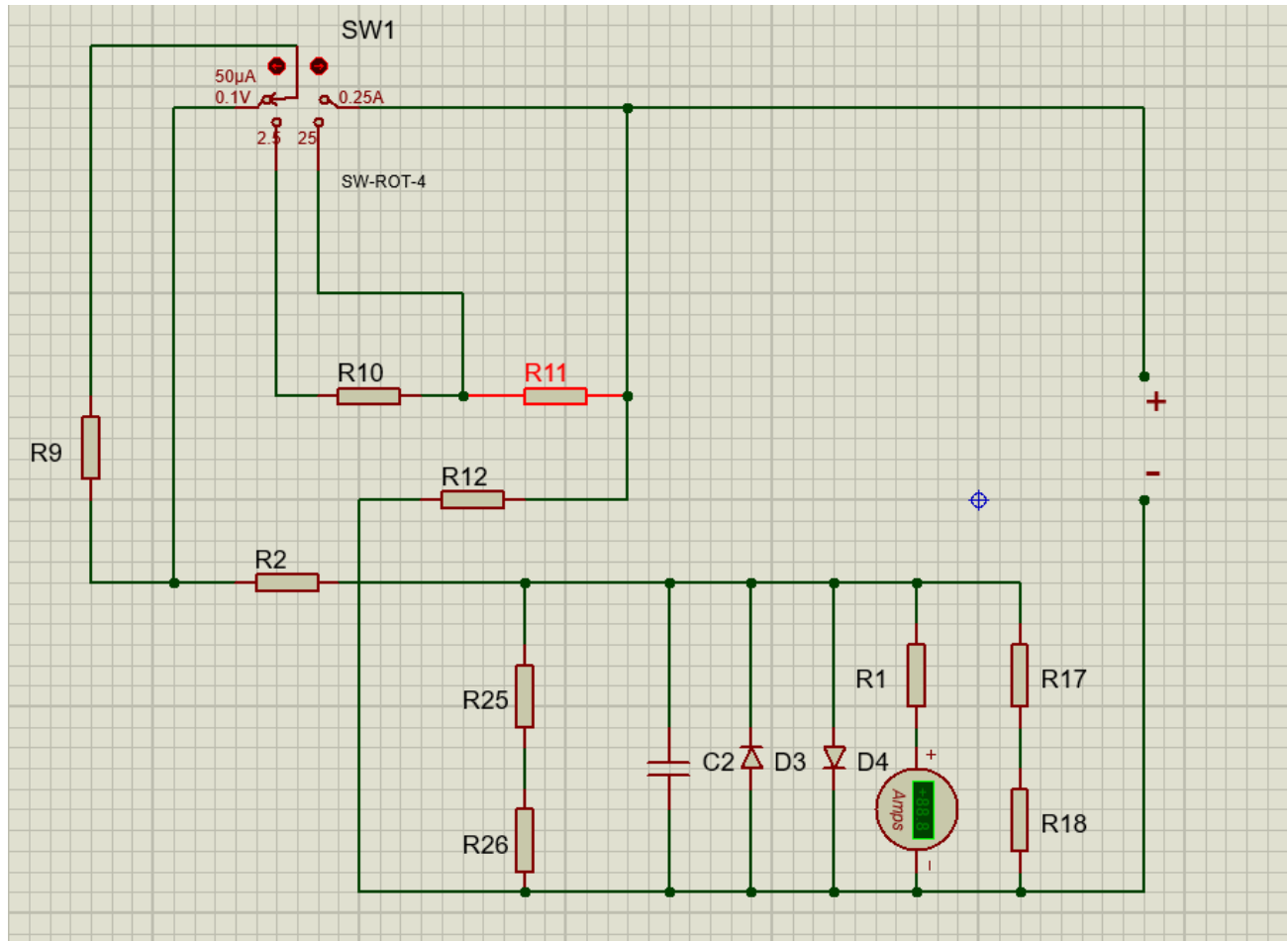


- Kích thước nhỏ gọn, kiểu dáng mạnh mẽ dễ dàng mang theo khi sử dụng hay bảo quản khi không sử dụng.
- Kết cấu chắc chắn, độ bền cao.
- Sản phẩm cho độ chính xác cao giúp bạn luôn yên tâm và có được hiệu quả cao trong công việc.
- Tự động tắt khi không sử dụng giúp tiết kiệm pin.
- Vỏ hộp có chức năng giống chân đế để bàn.
- Kết quả được đọc chính xác ở mặt đồng hồ kim.



-Hình 20: Đồng hồ Sanwa YX 360 TR

### 2.3: Sơ đồ mô phỏng trên Proteus



Hình 21: Sơ đồ mô phỏng mạch thang đo DC ma

### TÀI LIỆU THAM KHẢO

- [1].Giáo trình Kỹ thuật đo lường\_Trường đại học Giao thông vận tải
- [2].Slide bài giảng Kỹ thuật đo lường\_Bộ môn Kỹ thuật điện tử
- [3].Đồng hồ Elenco  
<https://www.elenco.com/wp-content/uploads/2017/10/M1250K-3.pdf>
- [4].Arduino  
<http://arduino.vn/>
- [5].Cảm biến DHT11