I, Giới thiệu về đề tài

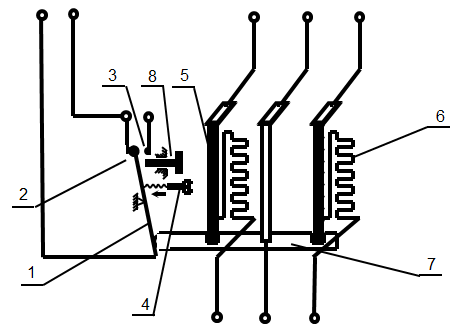
1.1 Đặt vấn đề

* Ngày nay nhu cầu công nghiệp số hóa các thiết bị đóng cắt, rơ le kỹ thuật số, … ngày càng phổ biến thông qua việc sử dụng các linh kiện bán dẫn, chip.
* Rơ le nhiệt bảo vệ quá tải và quá dòng là một thiết bị vô cùng quan trọng bảo vệ cho mạch điện tránh khỏi các sự cố quá tải và ngắn mạch gây hư hại lớn.
* Rơ le nhiệt có chức năng đóng cắt các tiếp điểm khi dòng điện tăng mạnh sinh ra nhiệt tác động lên thanh kim loại khiến chúng bị giãn nở ra. Nhờ sự có mặt của nó mà các thiết bị điện và máy móc sẽ hoạt động ổn định hơn cũng như không bị hư hỏng khi quá tải. Chính vì vậy, rơle nhiệt được ứng dụng trong hầu hết các hệ thống điện từ công nghiệp tới dân sự.
* Rơ le nhiệt sử dụng các chi tiết cơ khí và sư co giãn của thanh kim loại để phát hiện sự cố nên có thể bị ảnh hưởng lớn của môi trường xung quanh. Rơ le kỹ thuật số là thiết bị phát hiện sự cố về dòng điện bằng cách đo trực tiếp dòng điện của mạch nên không bị ảnh hưởng và tác động của môi trường xung quanh, ngoài ra rơ le kỹ thuật số còn có rất nhiều công dụng khác mà rơ le truyền thống không có như lưu trữ thông tin về các sự cố và thời gian xảy ra nó. Đó là một vấn đề vô cùng quan trọng cho việc kiểm tra và bảo trì hệ thống điện. Chính vì thế việc nghiên cứu và thiết kế ra rơ le kỹ thuật số bảo về quá dòng là một việc quan trọng và cấp thiết cho ngành điện và thiết bị điện- điện tử.

1.2 Đề bài yêu cầu.

- Ở đây đề bài sẽ là mô phỏng lại quá trình giãn nở ở thanh kim loại kép của rơ le nhiệt khi chịu dòng điện tăng lên quá dòng điện quá tải và ngắn mạch và co lại khi dòng điện trở lại giá trị an toàn, đồng thời sẽ tác động role bảo vệ mạch khi thanh kim loại kép tăng quá mức cho phép.

- Cụ thể quá trình tác động và quá trình ngắt như sau:



+ Quá trình tác động: khi dòng điện tăng cao(>Ip) nhiệt sinh ra lớn(I^2.R) làm thanh kim loại kép bị cong , thanh truyền B dịch chuyển sang phải, thanh truyền D dịch chuyển sang trái -> tiếp điểm thay đổi trạng thái

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đặc tính | A | B | p | Tr |
| Tác động chậm | 0.052 | 0.114 | 0.02 | 4.85 |
| Tác động nhanh | 19.61 | 0.491 | 2 | 21.6 |
| Tác động cực nhanh | 28.2 | 0.12 | 2 | 29.1 |

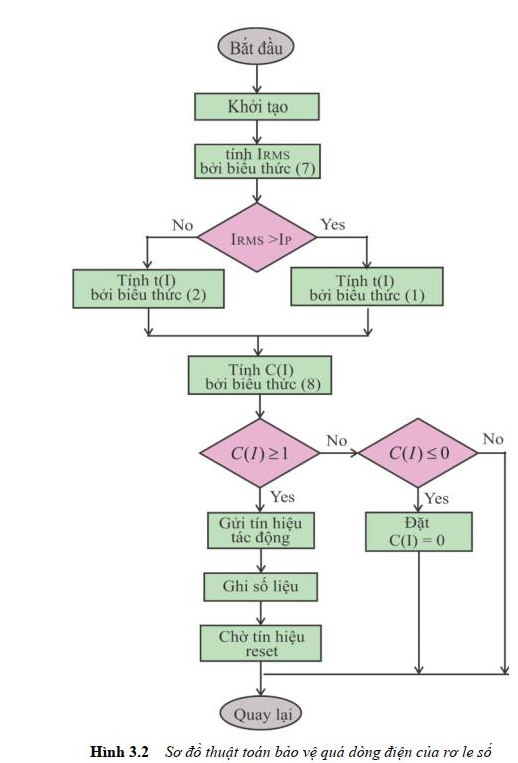
+ Quá trình ngắt: Khi dòng điện giảm(<Igh)nhiệt sinh ra nhỏ (I2R) hơn quá trình tỏa nhiệt của môi trường, là, thanh kim loại kép trở lại trạng thái bình , thanh truyền B dịch chuyển sang trai, thanh truyền D dịch chuyển sang phải -> tiếp điểm trở lại trạng thái bình thường.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Đặc tính | A | B | p | Tr |
| Tác động chậm | 0.052 | 0.114 | 0.02 | 4.85 |
| Tác động nhanh | 19.61 | 0.491 | 2 | 21.6 |
| Tác động cực nhanh | 28.2 | 0.12 | 2 | 29.1 |

* Ở sự cố ngắn mạch khi M>10 sẽ tác động rơ le ngay lập tức để bảo vệ mạch điện.

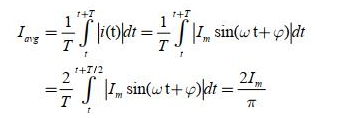
II, Giải pháp và phương pháp thực hiện.

* 1. Thuật toán:

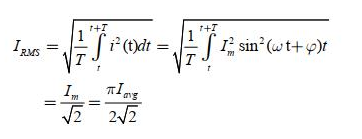


* 1. Phương pháp tính
     1. Tính dòng điện hiệu dụng.
* Gia trị hiệu dụng dòng điện (RMS) có thể tính toán dựa trên việc xử lý các mẫu giá trị thu được trên miền thời gian hoặc miền tần số. Bộ chính xác của giá trị thu được phụ thuộc vào bộ biến đổi ADC và phụ thuộc vào thuật toán xử lý tín hiệu. Ở đây em đề xuất việc tính toán giá trị RMS của dòng điện dựa trên miền thời gian, Đây là phương pháp đơn giản tiết kiệm bộ nhớ, hội tụ nhanh hơn sơ với các phương pháp khác.

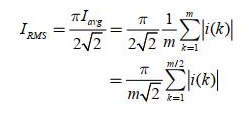
Phương pháp này dựa trên giả định dòng điện hình sin. Khi đó giá trị trung bình được xác định như sau:



Trong đó Im và T là các giá trị cực đại và chu kì của dòng điện. Từ đó ta suy ra giá trị hiệu dụng dòng điện như sau:



Để có thể triển khai được công thức trên ta cần để nó ở dạng rời rạc. Giả sử có m mẫu giá trị dòng điện i(k) được lấy mẫu từ bộ biến đổi ADC trong một chu kỳ dòng điện T.Khi đó Irms được xác định như sau:



Như vậy giá trị hiệu dụng sẽ được tính toán đơn giản và có thể áp dụng trong chip xử lý trung tâm.

* Sau khi tính giá trị hiệu dụng dòng điện Irms ta sẽ so sánh với giá trị dòng điện xác định quả tải và tính thực hiện các bước như nói ở phần trên(1.2) để xác định các trạng thái của dòng điện phát hiện ra các sự cố và tác động rơ le bảo vệ mạch.
  + 1. Thuật toán bảo vệ quá dòng cho rơ le.
* Thời gian tác động khi xảy ra sự cố quá tải có thể được xác định thông qua một hàm cộng dồn C(I):



Trong đó k chỉ số lần lặp của chương trình và Tc là hằng số chỉ thời gian giữa hai lần lặp liên tiếp của chương trình. C(I) tăng hoặc giảm từ 0 đến 1, khi C(I) tăng đến 1 sẽ tác động bảo vệ mạch.

* Rơ le chỉ có thể quay trở lại hoặt động bình thường khi có nủ reset được bấm.

III, Thiết kế phần cứng.

* Dựa trên các phân tích thuật toán và hướng giải quyết ở trên thì mạch thiết kế phần cứng của nhóm em để giải quyết bài toán được trình bày như sau, mạch gồm các khối chính sau:

3.1.1 Vi xử lý trung tâm Arduino Uno R3

3.1.2 Màn hình LCD16\*2 kết hợp module I2C PCF8574

-

3.1.3 Cảm biến dòng điện ACS712-20A đo dòng xoay chiều

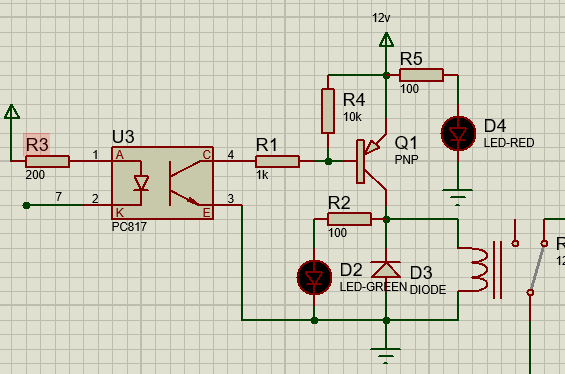
3.1.4 Module relay 12V + nguồn 12V

3.1.5 Tải thí nghiệm

3.1.6 Nút nhấn để reset

3.2 Thiết kế mạch phần cứng

3.2.1 Thiết kế khối đóng cắt relay 12V

- Khối đóng cắt rơ le 12V 1 kênh được thiết kế kích hoạt ở mức thấp gồm: IC opto quang PC817, transistor A1015 (PNP), relay 12V, diode, led báo trang thái nguồn, led báo trạng thái relay và các điện trở.

- Để kích hoạt được opto quang PC817 ta cần cấp dòng điện trong khoảng 10-25mA chạy từ cực anot đến cực katot của diot pháp quang bên trong IC. Từ đó em tính được điện trở R3 cần mắc là:

-Để hạn chế dòng điện chảy vào triac nội bên trong IC PC817 với dòng tối đa là 1A em mắc thêm điện trở R1 1000

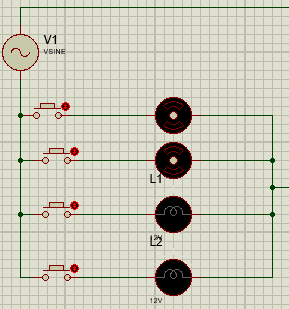
- Để hạn chế dòng vào bóng đèn báo tín hiệu với Imax=50mA em mắc thên điện trở R3 và R4 có giá trị 1k

= Tran A1015(PNP) có tác dụng khuếch đại dòng điện khi có tín hiệu điều khiển từ vi xử lý qua opto quang cách ly gửi đến chân Ib thì transistor sẽ dẫn cho dòng Ice đi qua dùng để kích hoạt relay

- Để Transistor có thể khuếch đại 1 tín hiệu điều khển dù rất nhỏ ta phân cực cho transistor bằng cách lắp thêm điện trở R4=10k để đặt transistor vào trạng thái sắn sàng hoạt động.

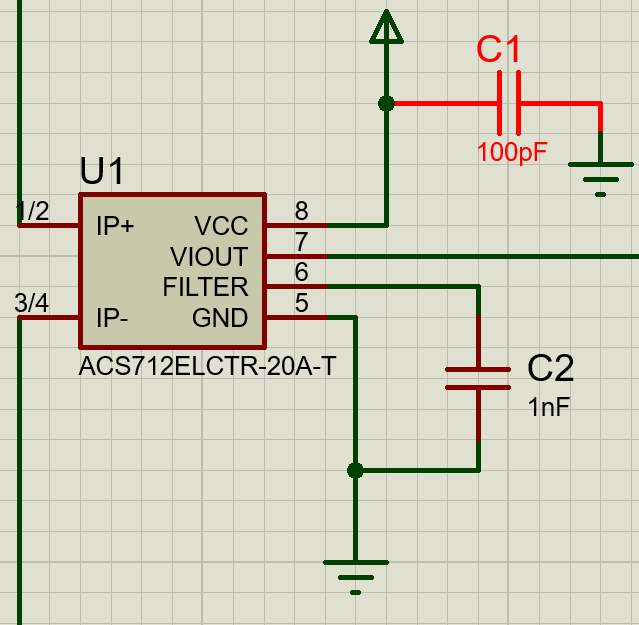
- Điode được lắp thêm để khép kín mạch với relay tiêu hao dòng khi có dòng quá độ xuất hiện để bảo vệ mạch relay khỏi bị phá hỏng do dòng lớn.

3.2.2 Thiết kế tải đo dòng.



- Tải mạch điện ở đây em dùng bộ 2 bóng đèn sợi đốt 40W và 2 quạt 30,40W được mắc song song và điều khiển bật tắt thông qua công tắc mục đích là để điều khiển dòng điện thay đổi trong mạch.Từ đó ta có thể tính tổng dòng điện trong mạch khi tất cả các tải được bật là:

3.3.3 Khối cảm biến dòng điện ACS712-20A

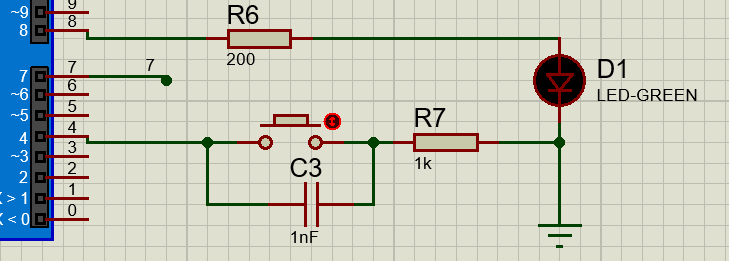


- Cảm biến ACS712\_20A được mắc nối tiesp 1 đầu với rơ le 12V 1 đầu với mạch tải song song để đo dòng điện trong mạch điện song song và tác động ngắt mạch điện ra qua rơ le khi xảy ra sự cố quá tải hoặc ngắn mạch để bảo vệ mạch điện.

- Hai chân Vcc và filter trên cảm biến sẽ được nối với tụ điện C1=100pF và C2=1nF và nối xuống đất để điện áp đầu đo cảm biến được ổn định hơn.

- Chân Viout được đưa ra để đo dòng điện cảm biến được và được nối đến chân A0 của vi điều khiển Arduino.

3.3.4 Thiết kế khối đèn báo hiệu và nút nhấn reset



- Khối nút nhấn và đèn báo hiệu nhằm reset lại mạch module relay khi xảy ra sự cố và báo hiệu bật đèn khi xảy ra sự cố sẽ phát sáng.

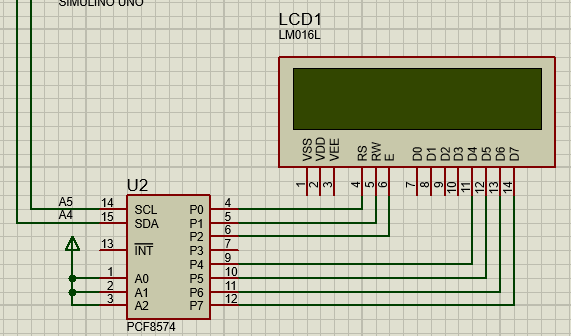
- Khối nút nhấn được thiết kế để đọc tín hiệu mức thấp gồm button,tụ điện 1nF mắc song song để chống dội phím và điện trở 1k để kéo xuống đất.

- Đèn báo hiệu khi xẩy ra sự cố được mắc với điện trở R6 để hạn chế dòng điện qua led với dòng điện qua led an toàn là 10-30mA từ đó ta xác định được ggias trị điện trở R6=5/25=200

3.2.5 Thiết kế khối hiển thị LCD

- Khối hiển thị LCD bao gồm màn hình hiển thị LCD16\*2 và mạch chuyển đổi I2C hỗ trợ LCD ( IC PCF8574) kết nối với vi điều khiển thông qua giao tiếp I2C

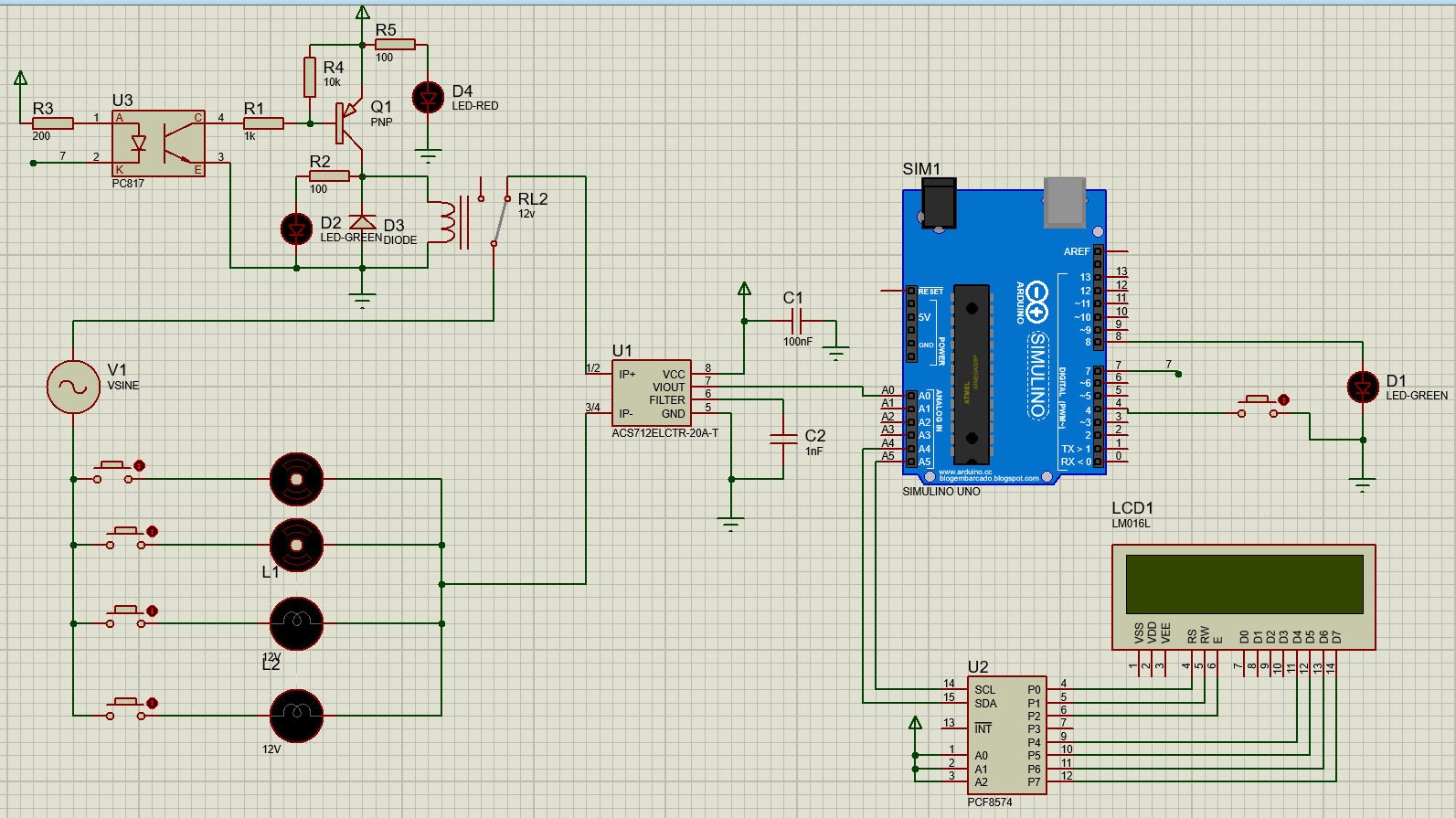
- Ưu điểm; Tiết kiệm chân cho VĐK



- Các chân A0,A1,A2 được kéo lên mức cao để xác định địa chỉ là 0x27

- SCL và SDA được nối với chân A5 và A4 cửa vi điều khiển đây là 2 chân SCL và SDA của arduino để giao tiếp với ngoại vi

3.2.6 Sơ đồ kết nối dây và sơ đồ nguyên lý tổng



*Hình : Sơ đồ kết nối các thiết bị*

IV, Thiết kế phần mềm và lập trình

4.1 Giới thiệu giao thức truyền thông và phần mềm lập trình sử dụng trong thiết kế

4.1.1 Giao thức truyền thông I2C

- Đầu năm 1980 Phillips đã phát triển một chuẩn giao tiếp nối tiếp 2 dây được gọi là I2C. I2C là tên viết tắt của cụm từ Inter-Intergrated Circuit. Đây là đường Bus giao tiếp giữa các IC với nhau. I2C mặc dù được phát triển bới Philips, nhưng nó đã được rất nhiều nhà sản xuất IC trên thế giới sử dụng. I2C trở thành một chuẩn công nghiệp cho các giao tiếp điều khiển, có thể kể ra đây một vài tên tuổi ngoài Philips như: Texas Intrument(TI), MaximDallas, analog Device, National Semiconductor ... Bus I2C được sử dụng làm bus giao tiếp ngoại vi cho rất nhiều loại IC khác nhau như các loại Vi điều khiển 8051, PIC, AVR, ARM... chip nhớ như: RAM tĩnh (Static Ram), EEPROM, bộ chuyển đổi tương tự số (ADC), số tương tự(DAC), IC điểu khiển LCD, LED...

- I2C sử dụng hai đường truyền tín hiệu:

* Một đường xung nhịp đồng hồ(SCL) chỉ do Master phát đi ( thông thường ở 100kHz và 400kHz. Mức cao nhất là 1Mhz và 3.4MHz).
* Một đường dữ liệu(SDA) theo 2 hướng.

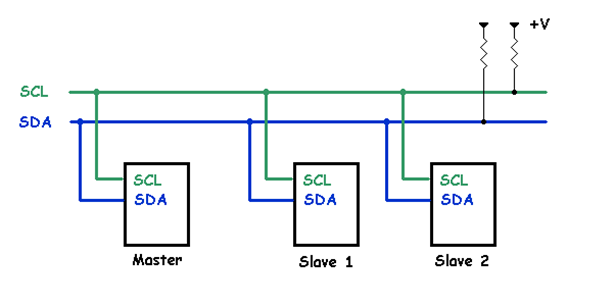
Có rất nhiều thiết bị có thể cùng được kết nối vào một bus I2C, tuy nhiên sẽ không xảy ra chuyện nhầm lẫn giữa các thiết bị, bởi mỗi thiết bị sẽ được nhận ra bởỉ một địa chỉ duy nhất với một quan hệ chủ/tớ tồn tại trong suốt thời gian kết nối. Mỗi thiết bị có thể hoạt động như là thiết bị nhận hoặc truyền dữ liệu hay có thể vừa truyền vừa nhận. Hoạt động truyền hay nhận còn tùy thuộc vào việc thiết bị đó là chủ (master) hãy tớ (slave).

Một thiết bị hay một IC khi kết nối với bus I2C, ngoài một địa chỉ (duy nhất) để phân biệt, nó còn được cấu hình là thiết bị chủ hay tớ.Tại sao lại có sự phân biệt này ? Đó là vì trên một bus I2C thì quyền điều khiển thuộc về thiết bị chủ. Thiết bị chủ nắm vai trò tạo xung đồng hồ cho toàn hệ thống, khi giữa hai thiết bị chủ-tớ giao tiếp thì thiết bị chủ có nhiệm vụ tạo xung đồng hồ và quản lý địa chỉ của thiết bị tớ trong suốt quá trình giao tiếp. Thiết bị chủ giữ vai trò chủ động, còn thiết bị tớ giữ vai trò bị động trong việc giao tiếp.

Về lý thuyết lẫn thực tế I²C sử dụng 7 bit để định địa chỉ, do đó trên một bus có thể có tới 2^7 địa chỉ tương ứng với 128 thiết bị có thể kết nối, nhưng chỉ có 112 , 16 địa chỉ còn lại được sử dụng vào mục đích riêng. Bit còn lại quy định việc đọc hay ghi dữ liệu (1 là write, 0 là read)

Điểm mạnh của I²C chính là hiệu suất và sự đơn giản của nó: một khối điều khiển trung tâm có thể điều khiển cả một mạng thiết bị mà chỉ cần hai lối ra điều khiển.

Ngoài ra I2C còn có chế độ 10bit địa chỉ tương đương với 1024 địa chỉ, tương tự như 7 bit, chỉ có 1008 thiết bị có thể kết nối, còn lại 16 địa chỉ sẽ dùng để sử dụng mục đích riêng (Mình chưa rõ lắm)

[](http://k3.arduino.vn/img/2016/07/31/0/2813_812450-1469971113-0-usi-conexao.png)

* Ở đây giao thức I2C được dùng để giao tiếp giữa vi điều khiển Arduino Uno với màn hình hiển thị LCD 16\*2 qua IC PCF8574 giúp mở rộng chân cho vi điều khiển.

4.1.2 Phần mềm lập trình cho vi điều khiển Arduino

- Sử dụng nền tảng lập trình Arduino IDE để lập trình cho vi điều khiển :

+ Do miễn phí và có cộng đồng phát triển lớn

+ Hỗ trợ nhiều thư viện để việc lập trình dễ dàng và nhanh hơn

4.1.3 Công cụ để phát triển phần cứng

- Sử dụng phần mềm thiết kế mạch proteus 8 để thiết kế mạch phần cứng cho thiết kế vì:

+ Proteus là phần mềm chuyên để thiết kế mạch phần cứng cho các ứng dụng làm mạch

+ Proteus dễ dàng sử dụng và thân thiện với sinh viên ngành điện

+ Có thể mô phỏng được để quan sát và thử nghiệm trước khi làm mạch thật.

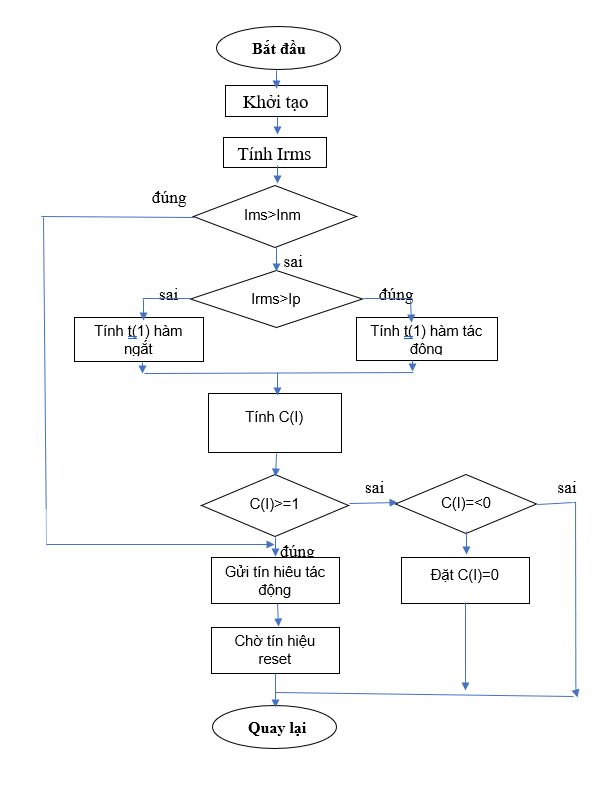
4.2 Chương trình cho vi điều khiển.

4.2.1 Lưu đồ thuật toán của chương trình

- Ở đây do thiết kế với mục đích để làm thí nghiệm nên em sẽ thay đổi các thông số về dòng điện quá tải và ngắn mạch để phân biệt dễ hơn, phù hợp với số lượng tải mà nhóm em sử dụng với dòng tải max là 0,682 và đảm bảo an toàn. Cụ thể :

+ Dòng quá tải ở đây e chọn là Ip=0,4(A)

+ Dòng ngắn mạch là Inm=0,6A

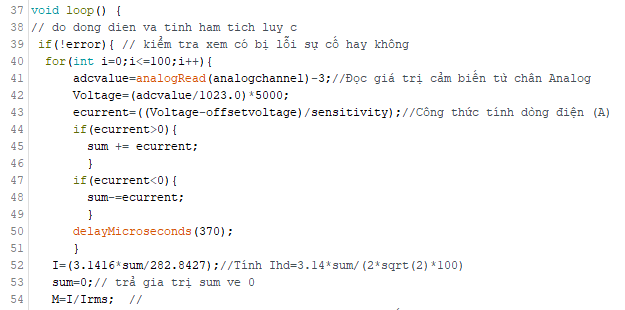
-Lưu đồ thuật toán: 

Chương trình code:

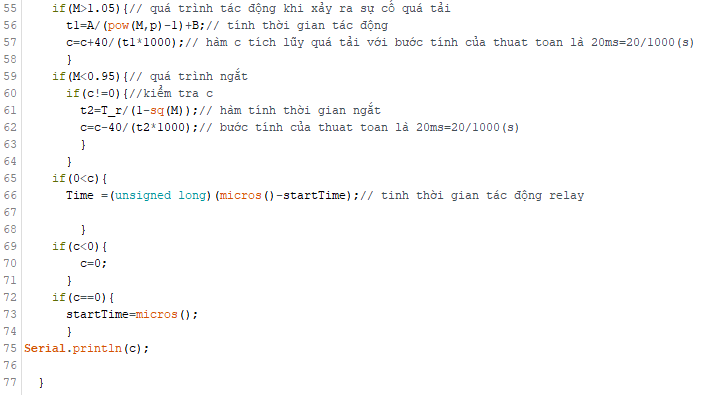
1. Khai báo các thư viện, các biến và hàm setup:



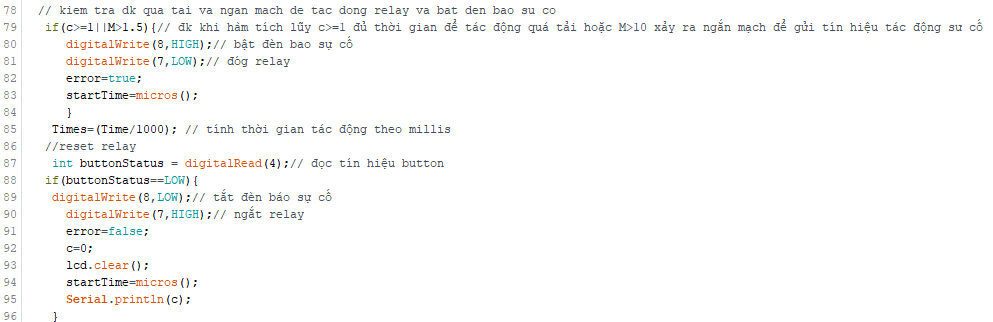
2.Đo dòng điện hiệu dụng và tính tỷ số M:



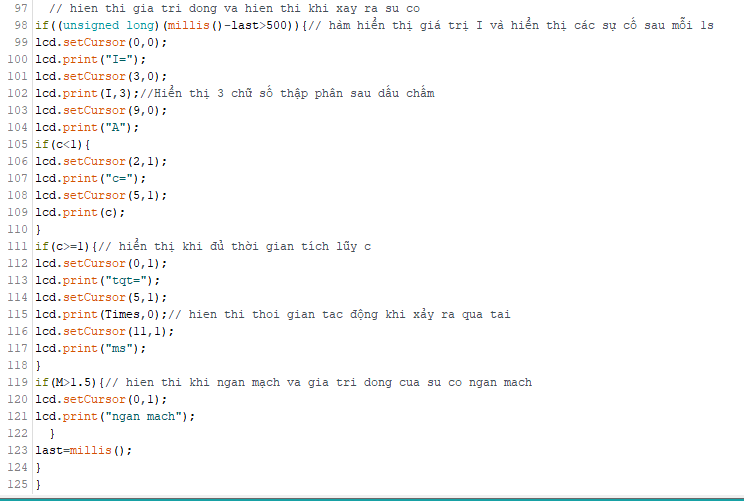
3.Tính C(I)



4.Kiểm tra điều kiện quá tải haowjc ngắn mạch để tác động relay và bật đèn báo sự cố.

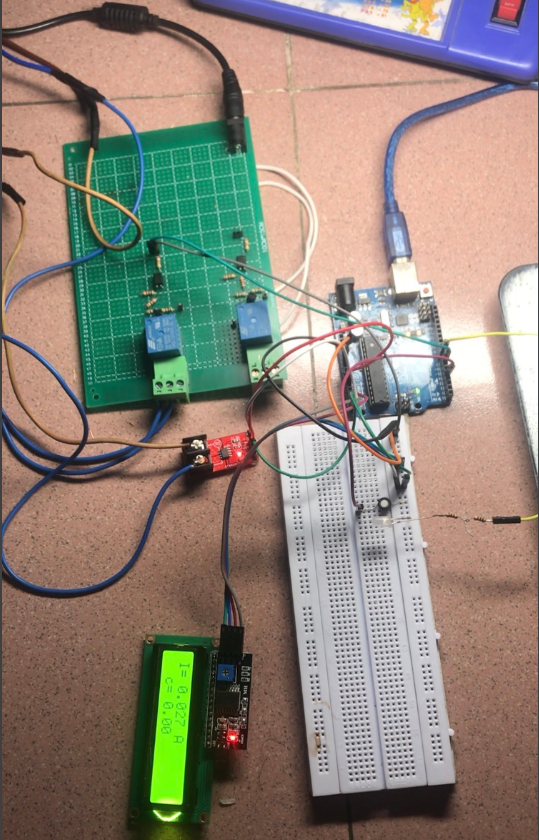


5. Hiển thị LCD giá trị dòng và hiển thị khi xảy ra sự cố.



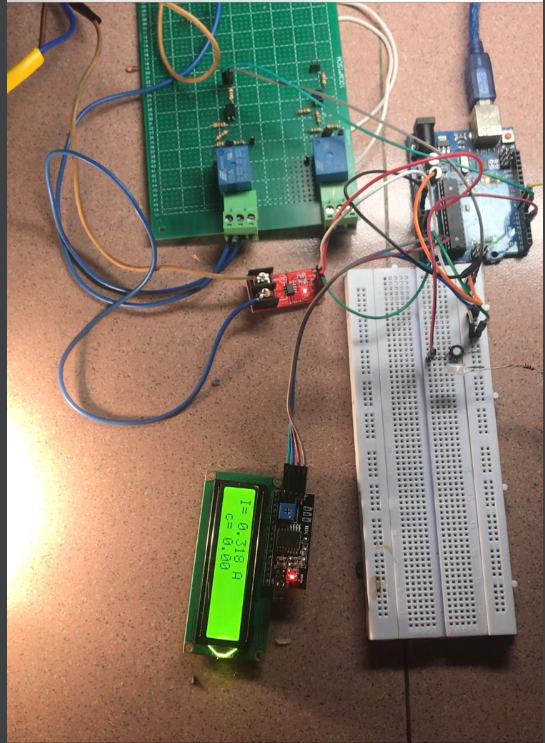
V, Kết quả thu được từ thực nghiệm

5.1 Sơ đồ mạch thật và đo dòng điện khi không tải,



* Ta có thể thấy khi không tải giá trị dòng điện đo được là 0,027A ~0A có sai số <5% nên có thể chấp nhận được

5.2 Đo dòng khi bật 2 đèn 40W.



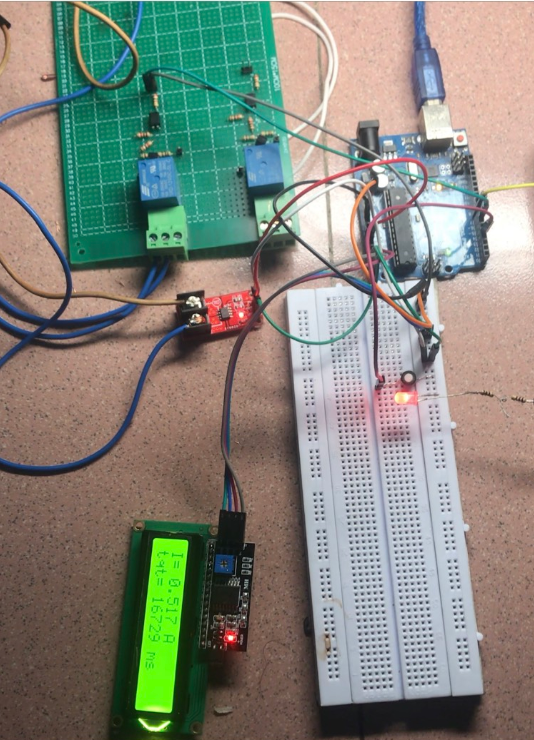
* Giá trị dòng điện ở đây đo được là Itt= 0,32A
* Theo lý thuyết ~Itt

5.3 Khi phát hiện sự cố quá tải tính hàm c(i) tính thời gian tác động và tác động relay bảo vệ mạch

- Khi bật thêm quạt có P=40W thì dòng điện tính theo lý thuyết trong mạch là:

~Itt=0,52

\_ Khi I=0,52>Ip=0,4 thì sẽ phát hiện sự cố quá tải và tính hàm tỷ lệ tích lũy c, c sẽ chạy và hiển thị trên lcd từ 0-1 khi đến 1 sẽ tác động relay, đèn báo sự cố sẽ sáng và hiển thị thời gian tác động

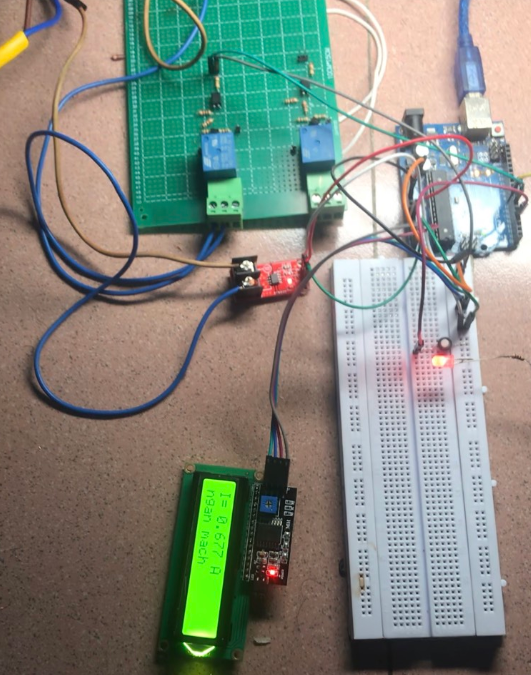


* Ở đây tqt=16729ms~16,7s gần đúng với thời gian thực tế đo được trên điện thoại từ khi bật thêm quạt 40W đến lúc tác động relay ngắt mạch là 16,3 s có thể chấp nhận đc.

5.4 Khi phát hiện sự cố ngắn mạch.

- Khi bật bật tất cả các thiết bị trong mạch gốm 2 đèn 40W và 2 quạt 30 và 40W thì ta tính được ~ Itt=0,677 đo được trên mạch thực tế

- Vì em đặt Inm=0,6 nên khi phát hiện I đo được > 0,6 thì mạch sẽ gửi tín hiệu tác động relay ngay lập tức để bảo vệ mạch điện, hiển thị báo lỗi sự cố ngắn mạch trên lcd đồng thời đèn báo sự cố sẽ sáng,



5.5 Kết luận

- Từ các kết quả thực tế trên ta có thể thấy sử dụng vi điều khiển để đo, phân tích tính toán sự cố có được độ chính xác cao đảm bảo có thể bảo vệ cho mạch điện an toàn khi sử dụng.

- Tuy nhiên vẫn còn sai số do tín hiệu đọc được từ vi điều khiển ADC 10bit chưa được chính xác tuyệt đối và mạch chưa gọn gàng.