

TRUNG TÂM SỬ CHỮA ĐIỆN TỬ QUẢNG BÌNH

MR. XÔ - 0901.679.359 - 80 Võ Thị Sáu, Phường Quảng Thuận, tx Ba Đồn, tỉnh Quảng Bình

GIÁ RẺ

NHANH CHÓNG

LINH KIỆN CHÍNH HÃNG



TRUNG TÂM SỬ CHỮA ĐIỆN TỬ XÔ NGUYỄN

- Dịch vụ sửa chữa điện tử tại nhà
- Cung cấp linh kiện điện tử
- Tư vấn lắp đặt nhà thông minh

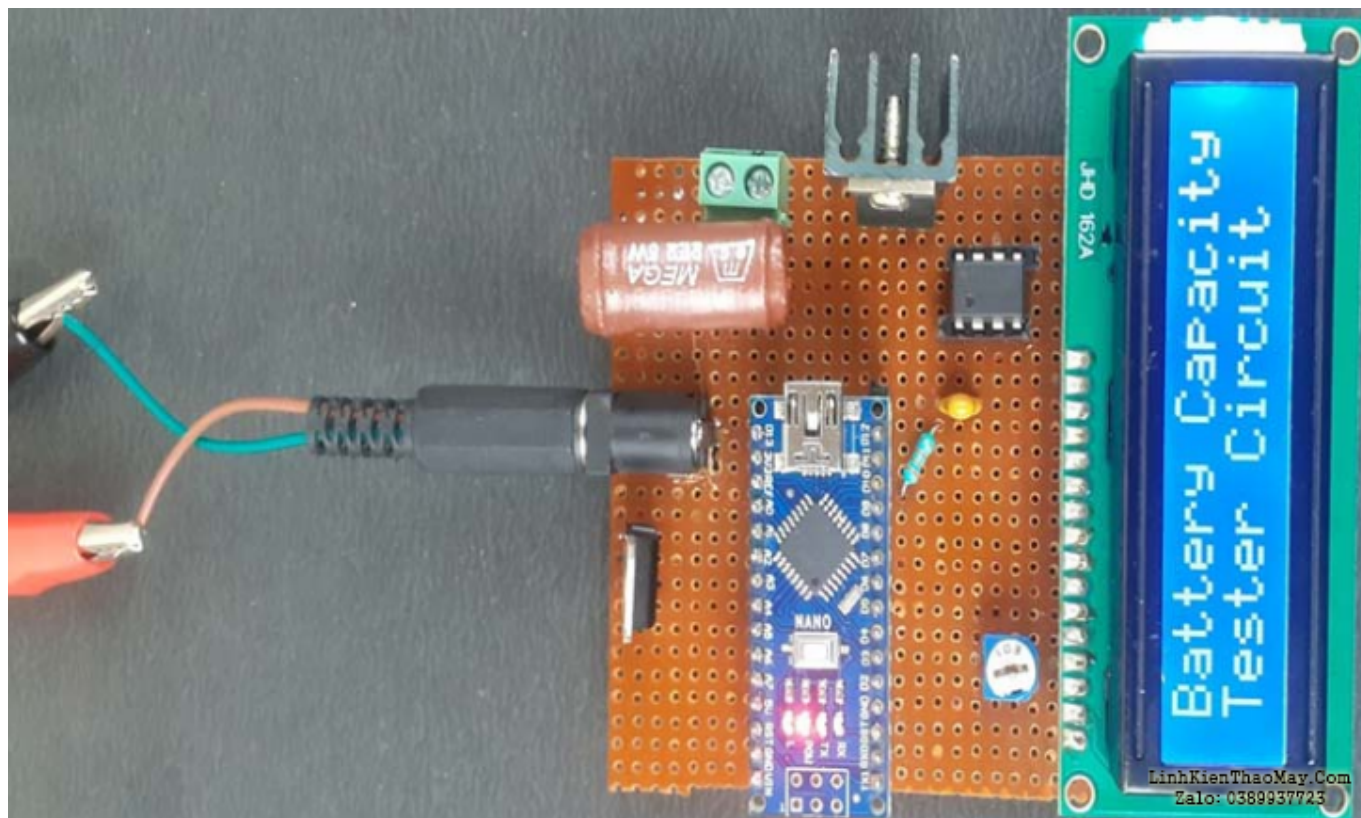
Đc: Quảng Thuận, tx Ba Đồn,
tỉnh Quảng Bình - 0901.679.359

Mạch kiểm tra dung lượng pin Lithium 18650 sử dụng Arduino : Với sự ra đời của công nghệ, các thiết bị và thiết bị điện tử của mình ngày càng trở nên nhỏ hơn với nhiều ứng dụng chức năng và phức tạp hơn. Với sự gia tăng độ phức tạp này, yêu cầu về nguồn điện của mạch cũng tăng lên và trong nhiệm vụ làm cho thiết bị càng nhỏ và di động càng tốt, mình cần một loại pin có thể cung cấp dòng điện cao trong một thời gian dài và đồng thời thời gian, cân nặng ít hơn nhiều để thiết bị vẫn di động

Trong số nhiều loại pin khác nhau hiện có, pin Axit Chì, pin Ni-Cd và pin Ni-MH không phù hợp vì chúng nặng hơn hoặc không thể cung cấp dòng điện cần thiết cho ứng dụng của mình, điều này khiến mình phải sử dụng pin lithium-ion có thể cung cấp dòng điện cao trong khi vẫn giữ được trọng lượng thấp và kích thước nhỏ gọn. Trước đây mình cũng đã xây dựng được một bộ sạc pin 18650 và module tăng cường và một hệ thống giám sát pin IOT dựa , bạn có thể kiểm tra xem chúng ra nếu quan tâm.

Tại sao mình cần một Mạch kiểm tra dung lượng pin Lithium 18650 ?

Có rất nhiều nhà cung cấp pin trên thị trường bán các phiên bản nhái giá rẻ của pin Li-ion tuyên bố các thông số kỹ thuật kỳ lạ với mức giá rất thấp, điều này là quá tốt để trở thành sự thật. Khi bạn mua các tế bào này, chúng hoàn toàn không hoạt động hoặc nếu có, dung lượng sạc hoặc dòng hiện tại quá thấp nên chúng không thể hoạt động với ứng dụng. Vậy **làm thế nào để kiểm tra pin lithium** nếu **pin** không phải là một trong những loại hàng nhái rẻ tiền này? Một trong những phương pháp là đo điện áp hở mạch khi không tải và có tải nhưng cách này không đáng tin cậy chút nào.



Vì vậy, mình sẽ xây dựng một **máy kiểm tra dung lượng pin 18650** cho Li-Ion 18650 Cell sẽ phóng điện một pin 18650 đã sạc đầy qua một điện trở trong khi đo dòng điện chạy qua điện trở để tính dung lượng của nó. Nếu bạn không thấy dung lượng pin đã xác nhận trong khi điện áp của tế bào nằm trong giới hạn quy định, thì tế bào đó bị lỗi và bạn không nên sử dụng vì Trạng thái sạc của tế bào sẽ cạn kiệt với tốc độ rất nhanh khi tải, tạo ra vòng lặp dòng điện cục bộ nếu được sử dụng trong bộ pin dẫn đến nóng và có thể cháy. Vì vậy, mình hãy nhảy ngay vào nó.

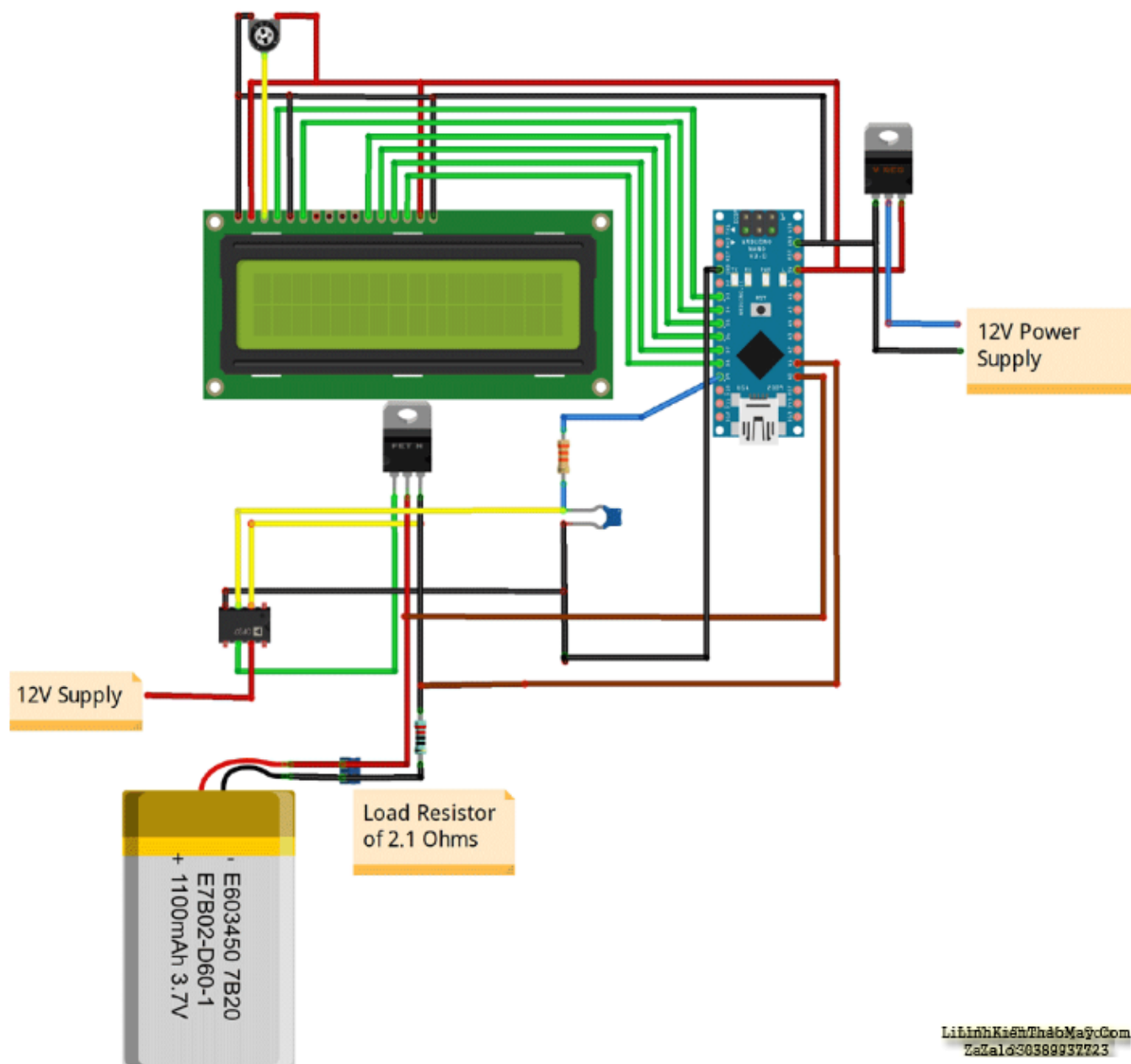
linh kiện bắt buộc

- Arduino Nano
- Màn hình LCD 16 × 2 ký tự
- IC OPAMP LM741
- Điện trở 2,2Ω, 5Watt
- 7805 IC điều chỉnh điện áp tích cực
- Nguồn điện 12V
- Chiết áp 10kΩ
- Tụ điện 0,47uF
- Điện trở 33kΩ
- Đầu nối giắc cắm nguồn DC
- Vít PCB
- IC Mosfet N-Kênh IRF540N
- Ván đục lỗ
- Bộ hàn
- Tản nhiệt



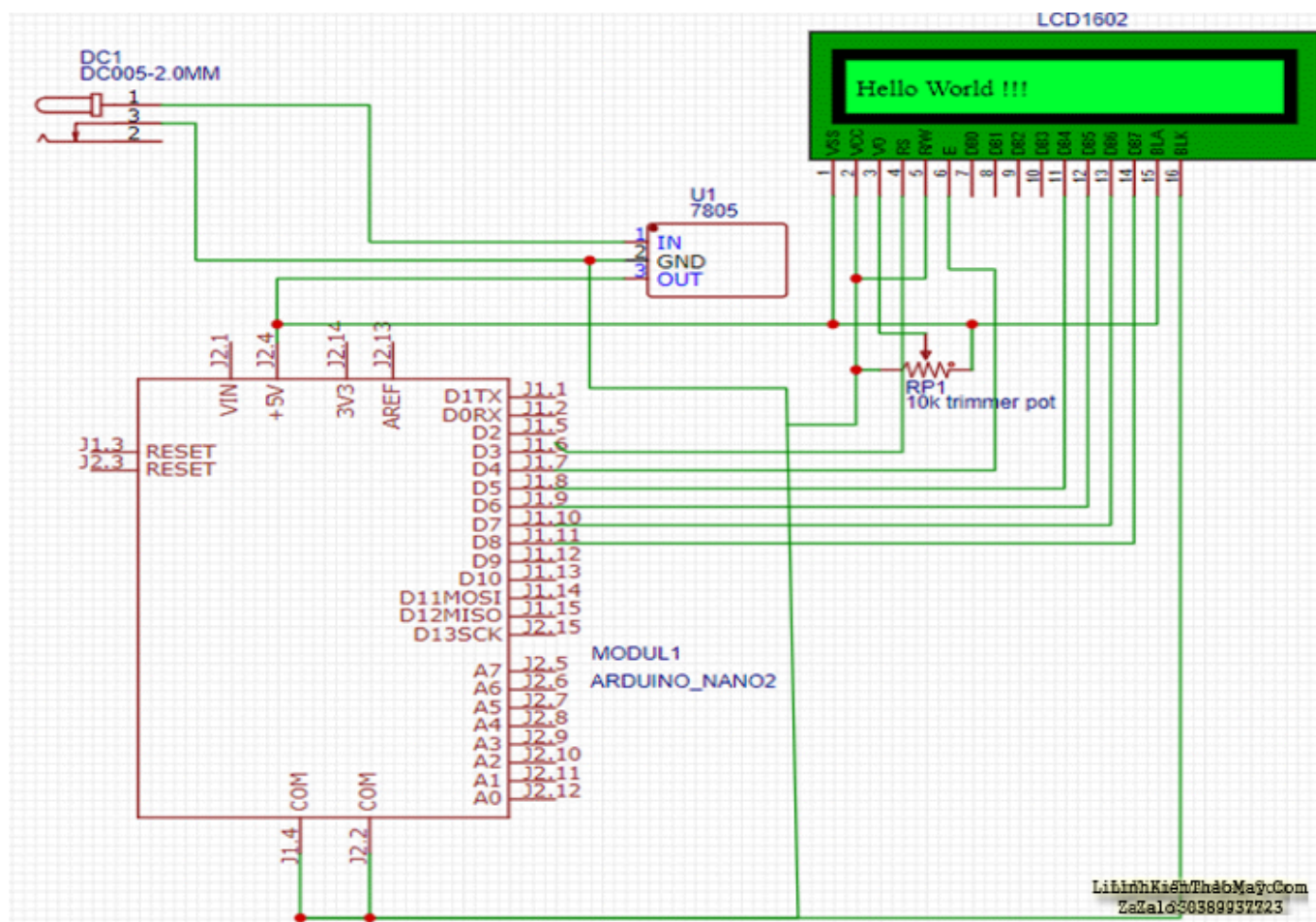
Sơ đồ Mạch kiểm tra dung lượng pin Lithium 18650

Sơ đồ mạch hoàn chỉnh cho **Mạch kiểm tra dung lượng pin Lithium 18650** được hiển thị bên dưới. Giải thích về mạch như sau-



Đơn vị tính toán và hiển thị:

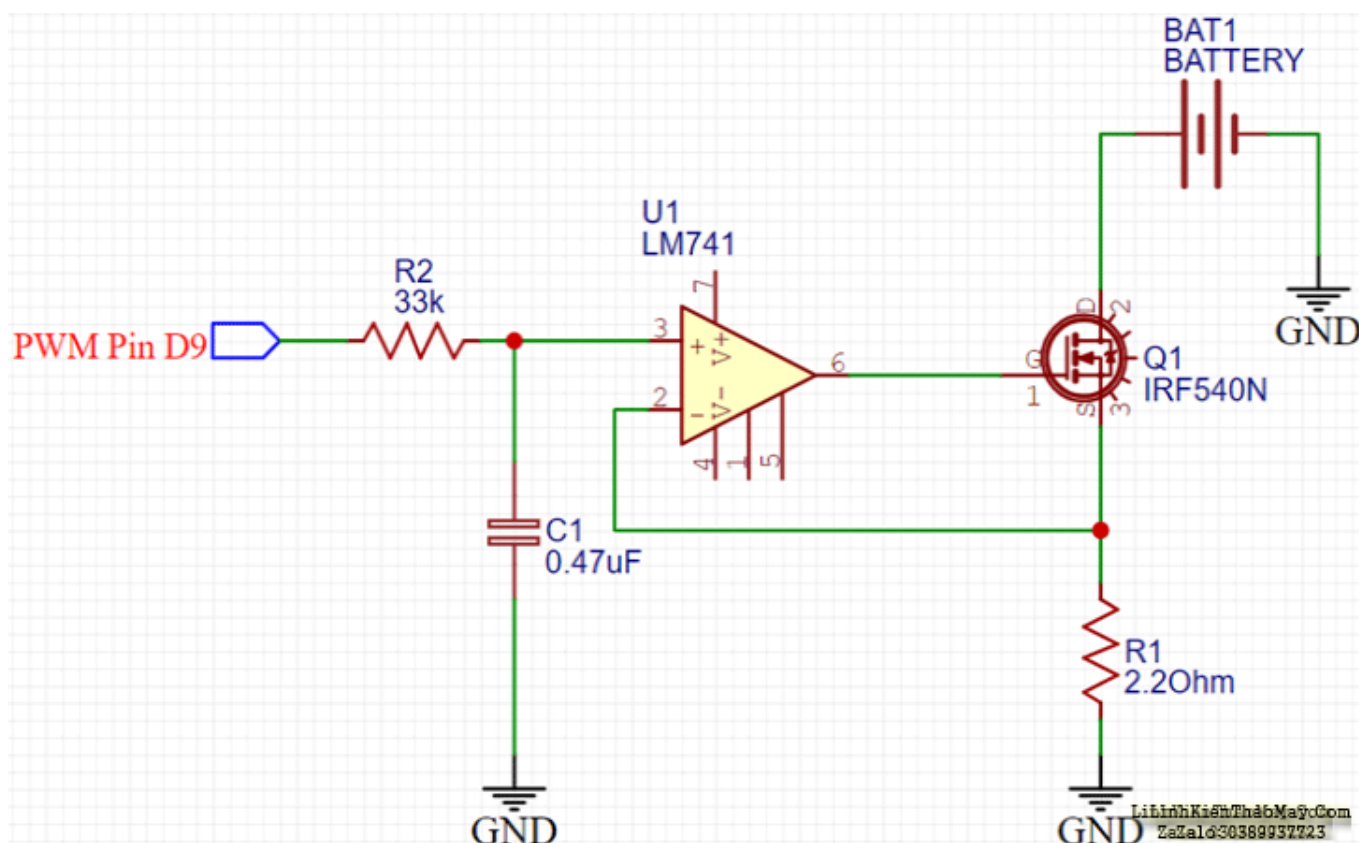
Mạch này được chia thành hai phần, đầu tiên là nguồn cung cấp 5V thấp cho Arduino Nano và màn hình LCD chữ và số 16×2 và các kết nối của chúng để hiển thị kết quả đo dòng điện và điện áp trong thời gian thực. Mạch được cung cấp bởi nguồn điện 12V sử dụng Nguồn xung SMPS hoặc bạn có thể sử dụng pin 12V cũng như dòng điện max sẽ vào khoảng 60-70mA để cấp nguồn cho Arduino và màn hình LCD.



Để giảm điện áp xuống 5V, mình sẽ sử dụng bộ điều chỉnh điện áp tuyến tính có thể lên đến 35V và cần nguồn điện đầu vào ít nhất 7,5V để cung cấp nguồn 5V được điều chỉnh và điện áp dư thừa sẽ bị tiêu tán dưới dạng nhiệt do đó nếu đầu vào của bạn điện áp IC ổn áp **LM7805** quá 12V thì cần nhắc thêm cục tản nhiệt để không bị hư. Màn hình LCD được cấp nguồn 5V từ 7805 và được kết nối với Arduino và hoạt động ở chế độ 4 bit. mình cũng đã thêm một **chiết áp gạt nước 10k Ω** để kiểm soát độ tương phản của màn hình LCD.

Mạch dòng tải không đổi:

Thứ hai là **mạch tải dòng điện không đổi** dựa trên PWM để làm cho dòng tải chạy qua điện trở được mình điều khiển và không đổi để không xảy ra lỗi do dòng điện thay đổi theo thời gian khi điện áp của tế bào giảm xuống. Nó bao gồm IC **OPAMP LM741** và **MOSFET kênh N IRF540N**, điều khiển dòng điện chạy qua MOSFET bằng cách BẬT và TẮT MOSFET theo mức điện áp do mình thiết lập.



Op-amp đang hoạt động ở **chế độ so sánh**, vì vậy trong chế độ này, đầu ra của op-amp sẽ cao bất cứ khi nào điện áp của chân không đảo ngược của op-amp cao hơn chân không đảo. Tương tự, nếu điện áp tại chân đảo ngược của op-amp cao hơn chân không đảo, đầu ra của op-amp sẽ bị kéo xuống. Trong mạch đã cho, mức điện áp chân không đảo ngược được điều khiển bởi chân D9 PWM của Arduino NANO, chân này chuyển đổi ở tần số 500Hz, sau đó được đưa qua bộ lọc mạch RC thông thấp với giá trị Điện trở 33kΩ và Tụ điện có điện dung 0,47 uF, để cung cấp tín hiệu DC gần như không đổi tại chân không đảo. Chốt đảo ngược được kết nối với điện trở tải, nó đọc điện áp trên điện trở và GND chung. Chân đầu ra của OPAMP được kết nối với cực cổng của MOSFET để BẬT hoặc TẮT. OPAMP sẽ cố gắng làm cho điện áp trên cả hai đầu của nó bằng nhau bằng cách chuyển mạch MOSFET được kết nối để dòng điện chạy qua điện trở sẽ tỷ lệ với giá trị PWM mà bạn đã đặt ở chân D9 của NANO. Trong Project này, dòng điện tối đa, mình đã giới hạn mạch của mình là 1,3A, điều này hợp lý vì tế bào mình có là 10A làm đỉnh mức dòng điện tối đa của nó

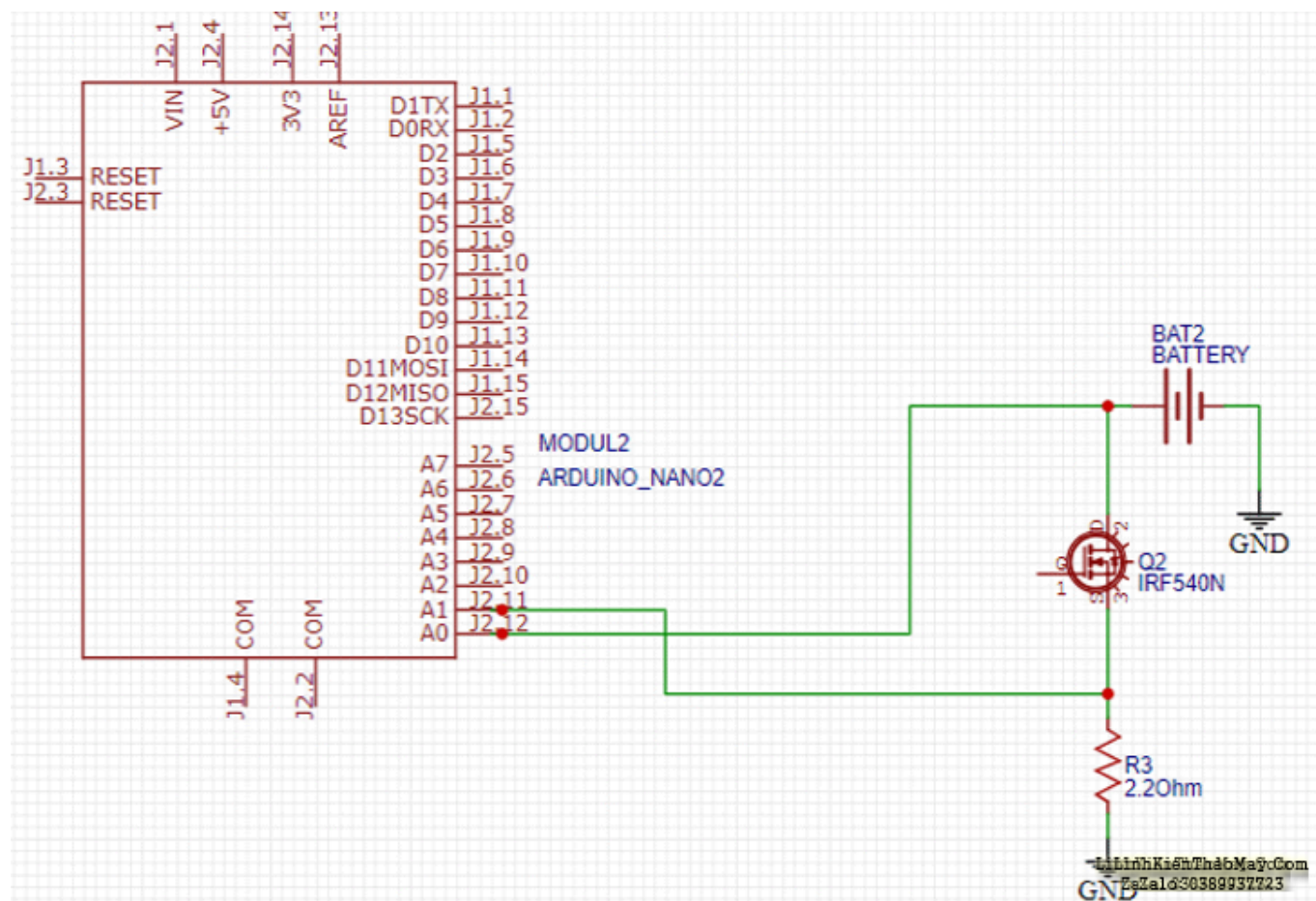
Đo điện thế:

Điện áp tối đa của một tế bào Li-Ion được sạc đầy điển hình là 4,1V đến 4,3V, thấp hơn giới hạn điện áp 5V của các chân đầu vào Analog của Arduino Nano, có điện trở bên trong hơn 10kΩ để mình có thể kết nối trực tiếp Di động đến các chân đầu vào analog nào mà không cần lo lắng về dòng điện chạy qua chúng. Vì vậy, trong Project này, mình cần đo điện áp của tế bào để có thể xác định xem tế bào có ở trong dải điện áp hoạt động chính xác hay không và nó đã được phóng điện hoàn toàn hay chưa.

mình cần đo dòng điện chạy qua điện trở vì mình không thể sử dụng shunt hiện tại vì độ phức tạp của mạch sẽ tăng lên và tăng điện trở trong đường dẫn tải sẽ làm giảm tốc độ

phóng điện của tế bào. Sử dụng điện trở shunt nhỏ hơn sẽ yêu cầu một mạch khuếch đại bổ sung để đọc điện áp từ nó, có thể đọc được đối với Arduino.

Vì vậy, mình đọc trực tiếp điện áp trên điện trở tải và sau đó sử dụng **Định luật Ohm chia** điện áp thu được cho giá trị điện trở tải để có dòng điện chạy qua nó. Đầu cực âm của điện trở được kết nối trực tiếp với GND, vì vậy mình có thể giả định một cách an toàn rằng điện áp mình đang đọc trên điện trở là điện áp giảm trong điện trở.



Chương trình Arduino để đo dung lượng pin

Bây giờ sau khi hoàn thiện mạch phần cứng, mình chuyển sang lập trình Arduino. Bây giờ nếu bạn chưa cài đặt Arduino IDE trên PC của mình thì bạn đang làm gì ở đây! Truy cập trang web chính thức của Arduino và tải xuống và cài đặt Arduino IDE hoặc bạn cũng có thể viết mã trong các trình soạn thảo nào khác nhưng đó là chủ đề cho một ngày khác vì bây giờ mình gắn bó với Arduino IDE. Bây giờ mình đang sử dụng Arduino Nano, vì vậy hãy đảm bảo rằng bạn đã chọn bảng Arduino Nano bằng cách đi tới **CÔNG CỤ > BOARD** và Chọn **ARDUINO NANO** ở đó, bây giờ hãy chọn đúng bộ xử lý nano của bạn bằng cách đi tới **CÔNG CỤ > BỘ XỬ LÝ** và trong khi bạn ở đó, hãy chọn cổng mà Arduino của bạn được kết nối trên PC. mình đang sử dụng Arduino để điều khiển màn hình LCD chữ và số 16 × 2 được kết nối với nó và để đo điện áp của tế bào và dòng điện chạy qua điện trở tải như đã giải thích trong phần trước, mình bắt đầu mã của mình bằng cách khai báo các tệp tiêu đề cho ổ đĩa 16 × 2 Màn hình LCD chữ và số. Bạn có thể bỏ qua phần này để nhận mã nấu chín và phục vụ đầy đủ ở cuối trang nhưng hãy chịu khó trong khi mình chia mã thành các phần nhỏ và cố gắng giải thích.

Bây giờ tệp tiêu đề đã được xác định, mình chuyển sang phân khai báo các biến, mình sẽ sử dụng trong đoạn mã để tính toán điện áp và dòng điện. Ngoài ra, mình phải xác định các chân mà mình đang sử dụng để điều khiển màn hình LCD và các chân mình sẽ sử dụng để cung cấp đầu ra PWM và đọc các điện áp tương tự đến từ cell và điện trở cũng trong phần này.

```
#include <LiquidCrystal.h> //Default Arduino LCD Librerey is included
const int rs = 3, en = 4, d4 = 5, d5 = 6, d6 = 7, d7 = 8; //Mention
the pin number for LCD connection
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
const float BAT_LOW = 3.0; //to define the low voltage limit of the
Li Ion cell
const float BAT_HIGH = 4.5; //to define the high voltage limit of
the cell
const int MOSFET_Pin=9;
const int PWM_VALUE=150;
unsigned long previousMillis = 0; // Previous time in ms
unsigned long millisPassed = 0; // Current time in ms
float Capacity=0; //Variable to define the battery Capacity
float Resistor=2.2; // Load Resistor Value is 2.2ohms
float mA;
```

Bây giờ đến phần thiết lập, Nếu bạn muốn luôn luôn kết nối Arduino với PC của mình và theo dõi tiến trình bằng cách sử dụng Serial Monitor và khởi chạy màn hình LCD tại đây. Nó cũng sẽ hiển thị thông báo chào mừng “Mạch kiểm tra dung lượng pin” trên màn hình trong 3 giây.

```
void setup()
{
  Serial.begin(9600);
  lcd.begin(16, 2);
  lcd.setCursor(0, 0); // Set the cursor on the first column and first
row.
  lcd.print("Battery Capacity");
  lcd.setCursor(0,1);
  lcd.print("Tester Circuit");
  delay(3000);
  lcd.clear();
}
```

Bây giờ mình không cần phải khai báo chân Arduino PWM làm Đầu ra vì hàm *AnalogWrite* mà mình sẽ sử dụng trong vòng lặp chính của mình sẽ đảm nhiệm phần này. Bạn cần xác định giá trị PWM sẽ được ghi trên chân đó trong mã. Chọn giá trị PWM một cách cẩn thận theo dòng phóng điện được yêu cầu trong ứng dụng của bạn. Giá trị PWM quá cao sẽ dẫn đến dòng điện cao với điện áp giảm cao trong cell Li-Ion và giá trị PWM quá thấp sẽ dẫn đến thời gian phóng điện cao của cell. Trong chức năng vòng lặp chính, mình sẽ đọc điện áp trên các chân A0 và A1 vì Arduino có bộ ADC 10 bit trên bo mạch, do đó mình sẽ

thấy các giá trị đầu ra kỹ thuật số nằm trong khoảng từ 0-1023 mà mình sẽ cần phải điều chỉnh trở lại Phạm vi 0-5V bằng cách nhân nó với **5.0 / 1023.0**. Đảm bảo rằng bạn đo chính xác điện áp giữa các chân 5V và GND của Arduino Nano bằng Vôn kế hoặc Đồng hồ vạn năng đã hiệu chỉnh vì hầu hết các trường hợp điện áp quy định không chính xác là 5,0V và thậm chí một sự khác biệt nhỏ trong điện áp tham chiếu này sẽ dẫn đến lỗi phát sinh trong các bài đọc điện áp, vì vậy hãy đo điện áp chính xác và thay thế 5,0 trong hệ số nhân đã cho ở trên.

Bây giờ để giải thích logic của mã, mình liên tục đo điện áp của ô và nếu điện áp của ô vượt quá giới hạn trên do mình chỉ định trong mã, thì thông báo lỗi sẽ hiển thị trên màn hình LCD để cho bạn biết liệu ô đó có quá tải hoặc có điều gì đó sai với kết nối và nguồn điện đến chân cổng MOSFET bị dừng để không có dòng điện nào có thể chạy qua điện trở tải. Điều quan trọng là bạn phải sạc đầy pin trước khi kết nối nó với bảng kiểm tra dung lượng để bạn có thể tính toán tổng dung lượng sạc của nó.

```
analogWrite(MOSFET_Pin, PWM_VALUE);
// read the input on analog pin 0:
int sensorValue_voltage_Cell = analogRead(A0);
// Convert the analog reading (which goes from 0 - 1023) to a
voltage (0 - 5V):
float voltage = sensorValue_voltage_Cell * (5.08 / 1023.0);
Serial.print("VOLTAGE: ");
Serial.println(voltage);           // Here the voltage is being printed
on Serial Monitor
lcd.setCursor(0, 0); // Set the cursor on the first column and first
row.
lcd.print("Voltage: "); // Print the voltage reading on the screen
lcd.print(voltage);
delay(100);
int sensorValue_Shunt_Resistor= analogRead(A1);
float voltage1= sensorValue_Shunt_Resistor *(5.08 / 1023.0);
float current= voltage1/Resistor;
Serial.print("Current: ");
Serial.println(current);
lcd.setCursor(0, 1); //Set the cursor on the first column and the
second row (counting starts at 0!).
lcd.print("Current: ");
lcd.print(current);
```

Bây giờ, nếu điện áp của tế bào nằm trong giới hạn điện áp trên và dưới do mình chỉ định thì Nano sẽ đọc giá trị Dòng điện theo phương pháp được chỉ định ở trên và nhân nó với thời gian đã qua trong quá trình đo và lưu trữ trong biến công suất mà mình đã xác định trước đó tính bằng đơn vị mAh. Trong toàn bộ thời gian này, các giá trị dòng điện và điện áp thời gian thực được hiển thị trên màn hình LCD được đính kèm, và nếu muốn, bạn cũng có thể xem chúng trên màn hình nối tiếp. Quá trình phóng điện của cell sẽ tiếp tục cho đến khi điện áp của cell đạt dưới mức giới hạn dưới do mình chỉ định trong chương trình và khi đó tổng công suất của cell được hiển thị trên màn hình LCD và dòng điện chạy qua điện trở được dừng lại

bằng cách kéo cổng MOSFET pin thấp.

```
else if(voltage > BAT_LOW && voltage < BAT_HIGH )
{ // Check if the battery voltage is within the safe limit
  millisPassed = millis() - previousMillis;
  mA = current * 1000.0 ;
  Capacity = Capacity + (mA * (millisPassed / 3600000.0)); // 1
Hour = 3600000ms to convert it into mAh units
  previousMillis = millis();
  delay(1000);
  lcd.clear();
}
```

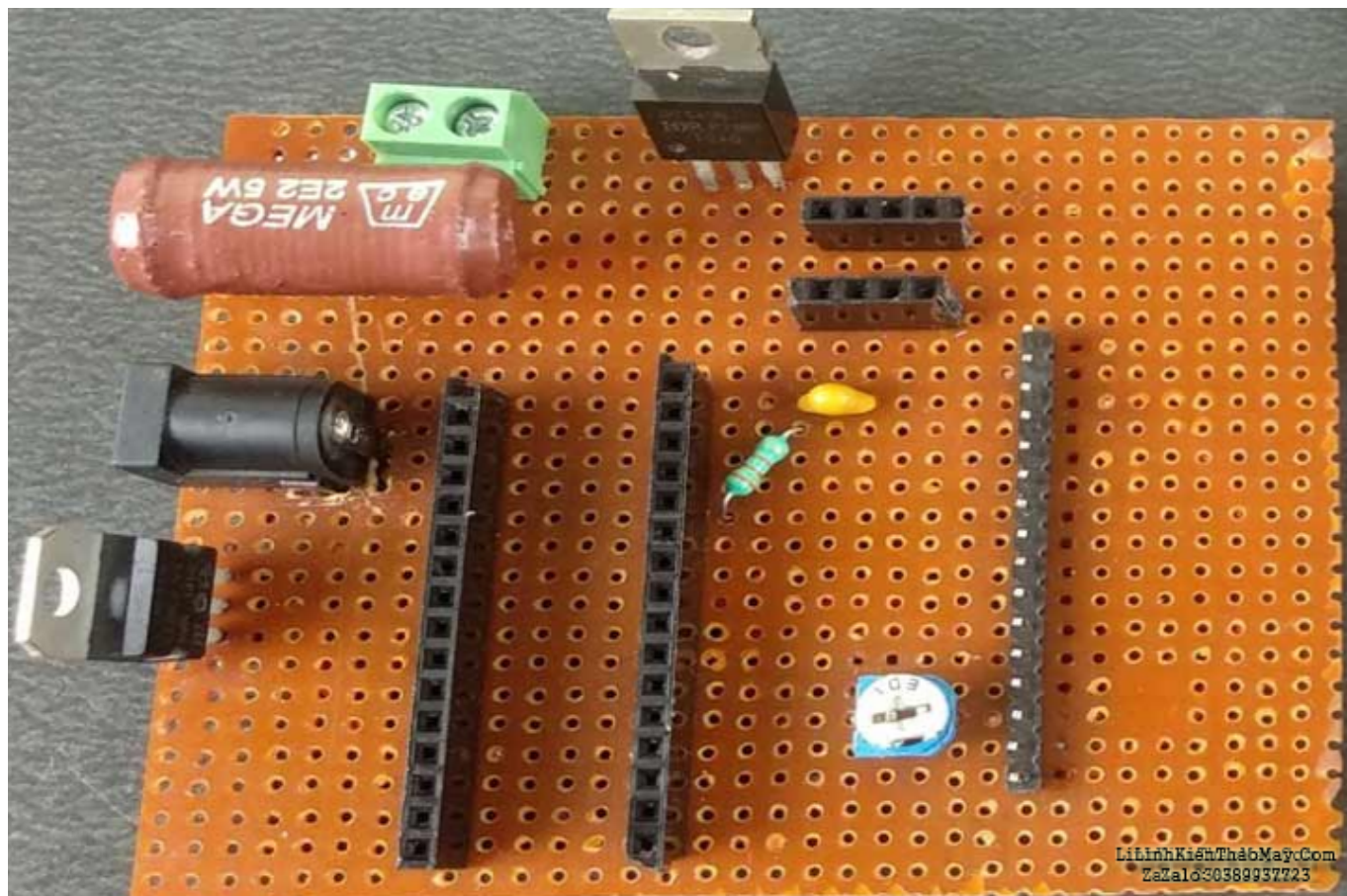
Cải tiến độ chính xác

Nó là một cách đủ tốt để đọc điện áp và dòng điện, nhưng nó không ok. Mối quan hệ giữa điện áp thực tế và điện áp ADC đo được không phải là tuyến tính và điều này sẽ dẫn đến một số sai số trong các phép đo điện áp và dòng điện.

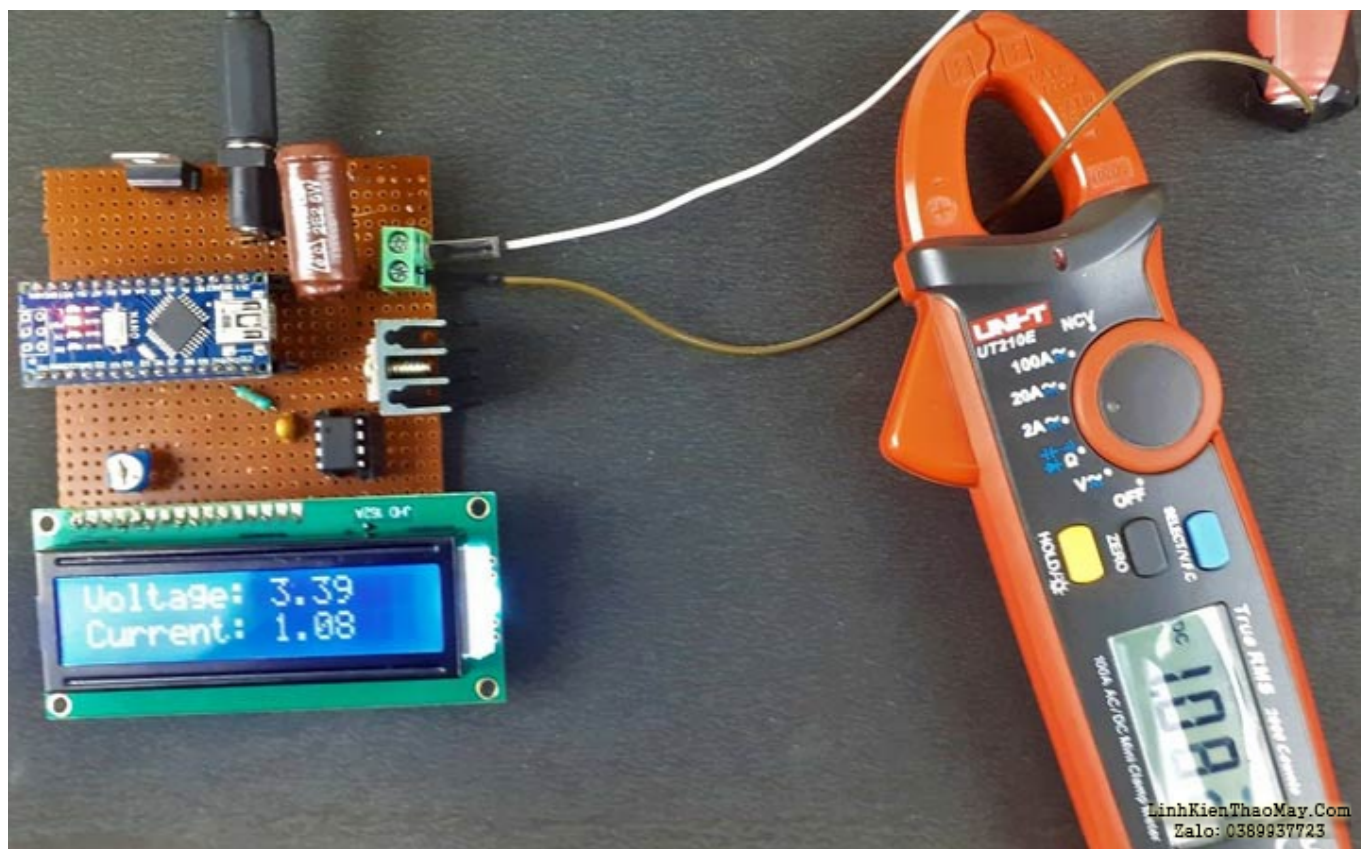
Nếu bạn muốn tăng độ chính xác của kết quả, thì bạn phải vẽ đồ thị các giá trị ADC bạn thấy từ việc áp dụng các nguồn điện áp đã biết khác nhau trên đồ thị và sau đó xác định phương trình hệ số nhân từ nó bằng cách sử dụng các phương pháp nào bạn thích. Bằng cách này, độ chính xác sẽ được cải thiện và bạn sẽ thấy rất gần với kết quả thực tế.

Ngoài ra, MOSFET mình sử dụng không phải là MOSFET mức logic, vì vậy nó cần hơn 7V để bật hoàn toàn kênh hiện tại và nếu mình áp dụng trực tiếp 5V vào nó, các số đọc hiện tại sẽ không chính xác. Nhưng bạn có thể sử dụng MOSFET N-Channel IRL520N mức logic để loại bỏ việc sử dụng nguồn cung cấp 12V và làm việc trực tiếp với mức logic 5V mà bạn có với Arduino của mình.

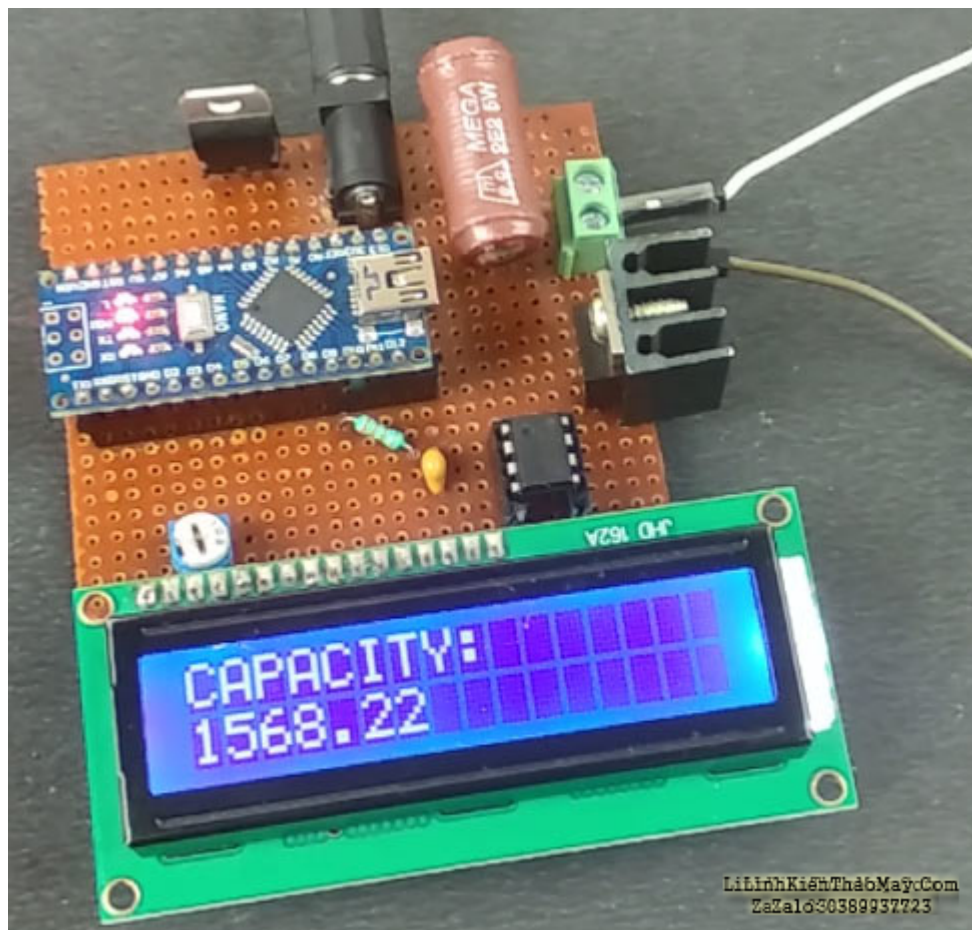
Xây dựng và kiểm tra mạch



Bây giờ khi mình thiết kế và thử các phần khác nhau của mạch trên bảng mạch và sau khi đảm bảo rằng tất cả chúng đều hoạt động như dự định, mình sử dụng Perfboard để hàn tất cả các linh kiện lại với nhau vì đây là một phương pháp chuyên nghiệp và đáng tin cậy hơn nhiều để kiểm tra mạch. . Nếu muốn, bạn có thể thiết kế PCB của riêng mình trên AutoCAD Eagle, EasyEDA, Proteus ARES hoặc các phần mềm nào khác mà bạn thích. Arduino Nano, LCD chữ và số 16 × 2 và OPAMP LM741 được gắn trên Female Bergstik để chúng có thể được sử dụng lại sau này.



mình đã cung cấp nguồn cung cấp 12V thông qua đầu nối DC Barrel Jack cho Mạch dòng tải không đổi và sau đó với sự trợ giúp của LM7805, nguồn 5V cho Nano và màn hình LCD được cung cấp. Bây giờ cấp nguồn cho mạch và điều chỉnh nổi tông đơ để đặt mức độ tương phản của màn hình LCD, lúc này bạn sẽ thấy Thông báo Chào mừng trên màn hình LCD, sau đó nếu mức điện áp của ô nằm trong phạm vi hoạt động thì dòng điện - điện áp và dòng điện từ pin sẽ được hiển thị ở đó.



Đây là một bài kiểm tra rất cơ bản để tính toán dung lượng của ô bạn đang sử dụng và nó có thể được cải thiện bằng cách lấy dữ liệu và lưu trữ nó trong tệp Excel để thực hiện xử lý và hiển thị dữ liệu bằng phương pháp đồ họa. Trong thế giới theo hướng dữ liệu ngày nay, đường cong phóng điện tế bào này có thể được sử dụng để xây dựng các mô hình dự đoán chính xác về pin nhằm mô phỏng và xem phản ứng của pin trong điều kiện tải mà không cần kiểm tra trong thế giới thực bằng cách sử dụng phần mềm như NI LabVIEW, MATLAB Simulink, v.v. và rất nhiều ứng dụng khác đang chờ bạn. Bạn có thể tìm thấy toàn bộ hoạt động của Project này trong video bên dưới. Nếu bạn có các câu hỏi nào về Project này, vui lòng viết chúng trong phần bình luận bên dưới hoặc sử dụng diễn đàn của mình. Hãy tận hưởng niềm vui với nó và nếu bạn muốn, mình có thể hướng dẫn bạn trong phần bình luận bên dưới về cách tiếp tục từ đây. Đến lúc đó Adios !!!

Tham khảo thêm : Mạch sạc Pin lithium tự ngắt

Code Mạch kiểm tra dung lượng pin Lithium 18650

```
#include <LiquidCrystal.h> //Default Arduino LCD Librarey is included
const int rs = 3, en = 4, d4 = 5, d5 = 6, d6 = 7, d7 = 8; //Mention the
pin number for LCD connection
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
const float BAT_LOW = 3.0; //to define the low voltage limit of the
Li Ion cell
const float BAT_HIGH = 4.5; //to define the high voltage limit of the
```



```
cell
const int MOSFET_Pin=9;
const int PWM_VALUE=50
;
unsigned long previousMillis = 0; // Previous time in ms
unsigned long millisPassed = 0; // Current time in ms
float Capacity=0; //Variable to define the battery Capacity
float Resistor=2.2; // Load Resistor Value is 2.5ohms
float mA;
void setup() {
// initialize serial communication at 9600 bits per second:
Serial.begin(9600);
lcd.begin(16, 2); //Initialise 16*2 LCD
lcd.setCursor(0, 0); // Set the cursor on the first column and first
row.
lcd.print("Battery Capacity");
lcd.setCursor(0,1);
lcd.print("Tester Circuit");
delay(3000);
lcd.clear();
}
void loop() {
analogWrite(MOSFET_Pin, PWM_VALUE);
// read the input on analog pin 0:
int sensorValue_voltage_Cell = analogRead(A0);
// Convert the analog reading (which goes from 0 - 1023) to a voltage
(0 - 5V):
float voltage = sensorValue_voltage_Cell * (5.12 / 1023.0)*1.2;
Serial.print("VOLTAGE: ");
Serial.println(voltage); // Here the voltage is being printed on
Serial Monitor
lcd.setCursor(0, 0); // Set the cursor on the first column and first
row.
lcd.print("Voltage: "); // Print the voltage reading on the screen
lcd.print(voltage);
delay(100);
int sensorValue_Shunt_Resistor= analogRead(A1);
float voltage1= sensorValue_Shunt_Resistor *(5.00 / 1023.0);
float current= voltage1/Resistor;
Serial.print("Current: ");
Serial.println(current);
lcd.setCursor(0, 1); //Set the cursor on the first column and the
second row (counting starts at 0!).
lcd.print("Current: ");
lcd.print(current);
if ( voltage > BAT_HIGH)
```

```
{
digitalWrite(MOSFET_Pin, LOW);      // Turned Off the MOSFET , No
discharge
Serial.println( "Warning High-V! ");
lcd.clear();
lcd.setCursor(0,0);
lcd.print("HIGH VOLTAGE!!");
delay(2000);
lcd.clear();
}
else if(voltage < BAT_LOW)
{
digitalWrite(MOSFET_Pin, LOW);      // Turned Off the MOSFET , No
discharge
Serial.println( "Warning Low-V! ");
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Low Voltage!!!");
delay(2000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("CAPACITY:");
lcd.setCursor(0,1);
lcd.print(Capacity);
delay(10000);
}
else if(voltage > BAT_LOW && voltage < BAT_HIGH )
{ // Check if the battery voltage is within the safe limit
millisPassed = millis() - previousMillis;
mA = current * 1000.0 ;
Capacity = Capacity + (mA * (millisPassed / 3600000.0)); // 1 Hour =
3600000ms to convert it into mAh units
previousMillis = millis();
//Serial.print("DATA,TIME,");      Serial.print(voltage);
Serial.print(","); Serial.println(Capacity);      //uncomment this
line to display on serial monitor
delay(1000);
lcd.clear();
}
}
```

Bài viết tham khảo :

- Mạch sạc pin 18650 Li-Ion bằng năng lượng mặt trời
- Mạch sạc pin axit chì

Các bài viết tương tự:

1. [Asus k43e - Để pin bật nguồn k lên. dùng adapter trực tiếp thỳ lên. Lắp pin và dùng adapter vẫn lên nhưng k có đèn báo pin. E nghi là hư pin phải k các bác. Cảm ơn các bác nhiều](#)
2. [bếp từ blustone - cắm điện kêu chuông bình thường đèn báo nguồn nháy chờ bấm phím,,,nhưng bấm on/off ko dc,,,kiểm tra các phím tốt đường mạch in tốt,,,cáp kết nối tốt,,,áp 5v đủ,,,các đường cáp AD,5V,DIO,CLK.STB GND,,,mạch dùng trở dán chủ yếu kiểm tra trở tốt,,,khoảng 1 phút sau báo lỗi E8 kèm chuông,,,,,](#)
3. [dạ em có con quạt hơi nước hiện tượng các nút ok riêng nút nguồn ko hư hỏng bấm ko tác dụng,,,khi bấm nút tắt ko tác dụng bấm nút này đèn led hiển thị của các nút yếu đi,,,mạch in dẫn tới nút ăn thẳng vào vi xử lý ko qua trở,,,,,em chưa kiểm tra nguồn - laoj quạt này\(quạt hơi nước\) cắm nguồn bấm nút chức năng số\(tốc độ\),hoặc quay hoặc hẹn giờ hoặc tạo âm vãn bình thường riêng nút tắt ko tắt dc,,,nguyên bản là tắt dc nhưng giờ là ko tắt dc](#)
4. [Đèn năng lượng mặt trời cho 1 cell pin 18650](#)
5. [Mạch sạc pin 18650 Li-Ion bằng năng lượng mặt trời](#)
6. [Mạch sạc pin Lithium-Ion sử dụng MCP73831](#)
7. [Máy sam sung 29z57 - Máy hư sò dòng em đã thay sò c5411 cắm vào nếu không cắm lái thì máy chạy cắm lái vào hư sò ngay em thay bằng sò c5144 thân to thì sò không hư máy không chạy cao áp kêu tạch tạch. Em kiểm tra lái không sao thay thử cao áp không được kiểm tra tất cả tụ c sò bằng đồng hồ đo tụ không có con nào hư. Hôm nay cắm lên kiểm tra thì máy tự dừng lại chạy cao áp réo sò nóng lên rất nhanh vậy mong các anh chỉ giúp](#)
8. [Phân tích mạch bảo vệ pin Lithium 3s 4s 5s \(chip CM1031, CM1041, CM1051\)](#)
9. [Phân tích mạch bảo vệ xả xả pin Lithium](#)
10. [Sạc Pin 18650 Li-Ion không chip](#)
11. [Tea2025b sử dụng với mạch stereo - Tự nhiên 1 bên của e k còn nghe thấy nữa e đã ktra kĩ hết đầu input ổn cả lúc sau thử thử cả 2 bên đều k thấy rì cả e đã thay 2 con 16v450uf nhưng vẫn bị.](#)
12. [tủ lạnh đông tuyết. \(tủ bảo ôn\) - bọc giàn. hết ga, mình đã lén giàn nóng riêng. và đã biết giàn nóng bị thủng. giờ mình muốn kiểm tra nguyên giàn lạnh xem có bị thủng ko mà mình ko nghĩ ra cách nào. vì mình mới vào nghề chưa am hiểu và chưa có kinh nghiệm j cả. vì cos một thợ trước đến nhà khách kiểm tra cái tủ này. ong thợ kia phán với chủ nhà là thủng giàn lạnh. giờ mình mới kiểm tra đc mỗi giàn nóng.](#)