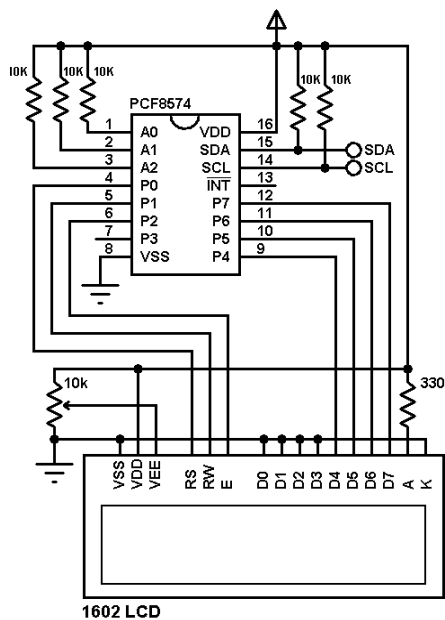


Bài 4: GIAO TIẾP I2C

I. MỤC ĐÍCH- YÊU CẦU:

- Giao tiếp I2C
- Viết chương trình điều khiển LCD dùng giao tiếp I2C
- Đọc thời gian thực giao tiếp I2C
- Đọc bàn phím cảm ứng dùng giao tiếp I2C

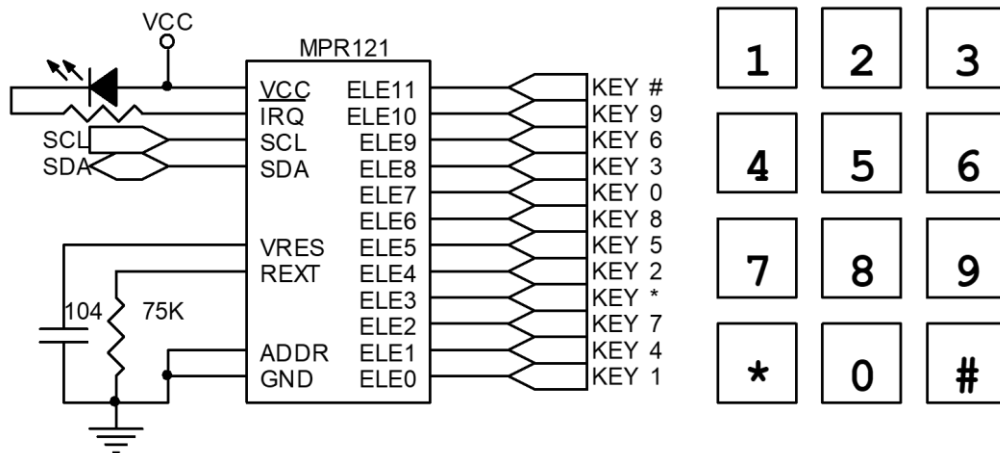
II. HIỂN THỊ LCD DÙNG GIAO TIẾP I2C



```
#include <Wire.h>
#include "LiquidCrystal_I2C.h"
LiquidCrystal_I2C lcd(0x27,16,2);
void setup()
{
    lcd.init(); //khởi tạo LCD
    lcd.backlight(); //bật đèn nền
    lcd.clear(); //xóa màn hình
    lcd.setCursor(0,0); //đến vị trí cột 0 của dòng 0
    lcd.print("STU");
    lcd.setCursor(0,1); //đến vị trí cột 0 của dòng 1
    lcd.print("FEEE");
}
void loop()
{
}
```

1. Bạn hãy khảo sát các hàm của thư viện **LiquidCrystal_I2C.h**
2. Viết chương trình hiển thị chữ DH CONG NGHE SAIGON ở hàng trên và hàng dưới hiển thị chữ KHOA ĐIỆN ĐIỆN TU
3. Điều chỉnh chương trình để dòng chữ trên dịch chuyển từ trái qua phải.
4. Viết chương trình hiển thị bộ đếm từ 00 đến 99 trên dòng 2 của LCD.

III. ĐỌC BÀN PHÍM CẢM ỨNG



```
#include <Wire.h>
#include <Adafruit_MPR121.h>
Adafruit_MPR121 cap = Adafruit_MPR121();
uint16_t lasttouched = 0;
uint16_t currouched = 0;

void setup() {
  Serial.begin(9600);
  Serial.println("Chương trình test phim ma tran giao tiep I2C");
  if (!cap.begin(0x5A)) {
    Serial.println("MPR121 không tìm thấy");
    while (1);
  }
  Serial.println("Đã thay chip MPR121");
}

void loop()
{
  // đọc trạng thái các phím cảm ứng
  currouched = cap.touched();
  for (uint8_t i=0; i<12; i++) {
    // nếu trạng thái hiện tại nhấn và trước đây chưa nhấn
    if ((currouched & _BV(i)) && !(lasttouched & _BV(i)) ) {
      Serial.print(i); Serial.println("Press");
    }
    // nếu trạng thái trước đây nhấn và hiện tại không nhấn
    if (!(currouched & _BV(i)) && (lasttouched & _BV(i)) ) {
      Serial.print(i); Serial.println(" Release");
    }
  }
  // cập nhật trạng thái trước đó.
  lasttouched = currouched;
}
```

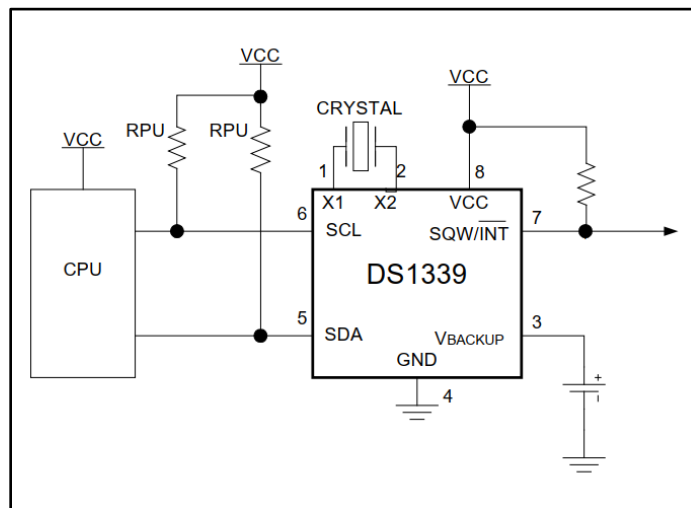
```

delay(100);
}

```

1. Cài đặt thư viện liên quan đến bàn phím cảm ứng **Adafruit_MPR121**
2. Thử nghiệm chương trình mẫu trên, giải thích hoạt động của chương trình.
3. Viết chương trình hiển thị các phím bấm cảm ứng lên màn hình LCD, các số được hiển thị dần từ phải qua trái, bấm phím '*' để xuống dòng, bấm phím '#' để xóa hiển thị, đưa con trỏ về đầu dòng.

IV. ĐỌC THÔNG SỐ THỜI GIAN THỰC



```

#include <Wire.h>
#include "LiquidCrystal_I2C.h"
#include <DS1337.h>
LiquidCrystal_I2C lcd(0x27,16,2);
struct time_t time; // struct chứa thời
void setup()
{
    lcd.init();
    lcd.backlight();
    lcd.clear();
}
void loop()
{
    ds1337_read_time(&time);
    lcd.setCursor(0,1);
    lcd.print(String(time.minute)+" Phut "+(time.second)+" Giay "); //chuyển
    từ số sang chuỗi
    delay(500);
}

```

1. Thực hiện chương trình mẫu trên, giải thích hoạt động của chương trình

2. Viết chương trình hiển thị thông số thứ ngày tháng năm trên dòng 1 và giờ phút giây ở dòng 2 của LCD.
3. Viết chương trình để thiết lập lại thông số thời gian của chip thời gian cho đúng với thời gian hiện tại.
4. Viết chương trình kết hợp hiển thị LCD , đọc phím cảm ứng và đọc thời gian thực để hiển thị thời gian trên LCD và cho phép dùng phím cảm ứng để thiết lập thời gian của chip thời gian thực.