

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN  
KHOA KỸ THUẬT MÁY TÍNH**



**BÁO CÁO MÔN HỌC  
THIẾT KẾ HỆ THỐNG NHÚNG  
CE224.N13.MTCL**

**GIẢNG VIÊN HƯỚNG DẪN: ĐOÀN DUY  
SINH VIÊN THỰC HIỆN: VÕ MINH MÃN – 20521594  
NGUYỄN ĐỨC DUY KHANG – 20521435  
TRƯƠNG TRẦN HOÀI NAM – 20521638**

**TP. HỒ CHÍ MINH, 12/2022**

# BÁO CÁO THIẾT KẾ HỆ THỐNG NHÚNG CE224.N13

Môn học: THIẾT KẾ HỆ THỐNG NHÚNG- Mã lớp: CE224.N13

Giảng viên hướng dẫn: Đoàn Duy

<b>Đề tài</b>	Điều khiển tốc độ động cơ dựa trên cường độ ánh sáng
<b>Thông tin sinh viên</b>	Mã số sinh viên: 20521594 Họ và tên: Võ Minh Mẫn Mã số sinh viên: 20521435 Họ và tên: Nguyễn Đức Duy Khang Mã số sinh viên: 20521638 Họ và tên: Trương Trần Hoài Nam
<b>Link các tài liệu tham khảo (nếu có)</b>	<a href="https://www.instructables.com/Arduino-LCD-Display-FlyIn-Text-Effect/">https://www.instructables.com/Arduino-LCD-Display-FlyIn-Text-Effect/</a> <a href="http://arduino.vn/reference/attachinterrupt">http://arduino.vn/reference/attachinterrupt</a> <a href="https://arduino.vn/giao-tiep-i2c-lcd-arduino/">https://arduino.vn/giao-tiep-i2c-lcd-arduino/</a> <a href="http://arduino.vn/reference/xung-pwm">http://arduino.vn/reference/xung-pwm</a>
<b>Đánh giá của giảng viên:</b> + Nhận xét + Các lỗi trong chương trình + Gợi ý	

## MỤC LỤC

<b>PHẦN 1: CÁC LINH KIỆN SỬ DỤNG TRONG MẠCH.....</b>	<b>1</b>
1.1 <i>Arduino UNO.....</i>	<i>1</i>
1.2 <i>Các linh kiện khác.....</i>	<i>2</i>
1.3 <i>Sơ đồ mạch.....</i>	<i>3</i>
<b>PHẦN 2: Ý TƯỞNG .....</b>	<b>3</b>
<b>PHẦN 3: CODE.....</b>	<b>4</b>
3.1 <i>Intro ENGINE CONTROL.....</i>	<i>4</i>
3.2 <i>MODE.....</i>	<i>5</i>
<b>PHẦN 4: ƯU VÀ NHƯỢC ĐIỂM CỦA HỆ THỐNG .....</b>	<b>8</b>
4.1 <i>Ưu điểm.....</i>	<i>8</i>
4.2 <i>Nhược điểm.....</i>	<i>8</i>

## DANH MỤC HÌNH

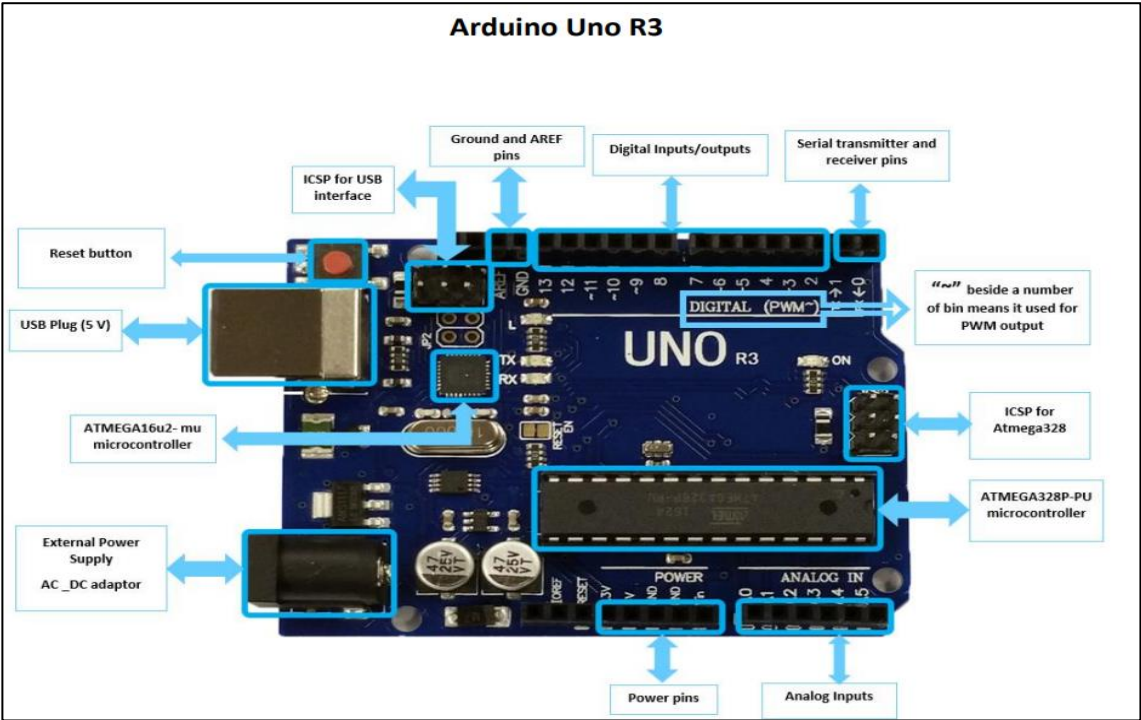
Hình 1.1 Arduino Uno R3.....	1
Hình 1.2 Sơ đồ chân Ardunio Uno R3.....	2
Hình 1.3 Sơ đồ mạch hoàn thiện.....	3

# DANH MỤC BẢNG

Bảng 1.1 Liệt kê các thông số cơ bản của Ardunio .....	1
Bảng 1.2 Thông tin các linh kiện khác.....	2

# PHẦN 1: CÁC LINH KIỆN SỬ DỤNG TRONG MẠCH

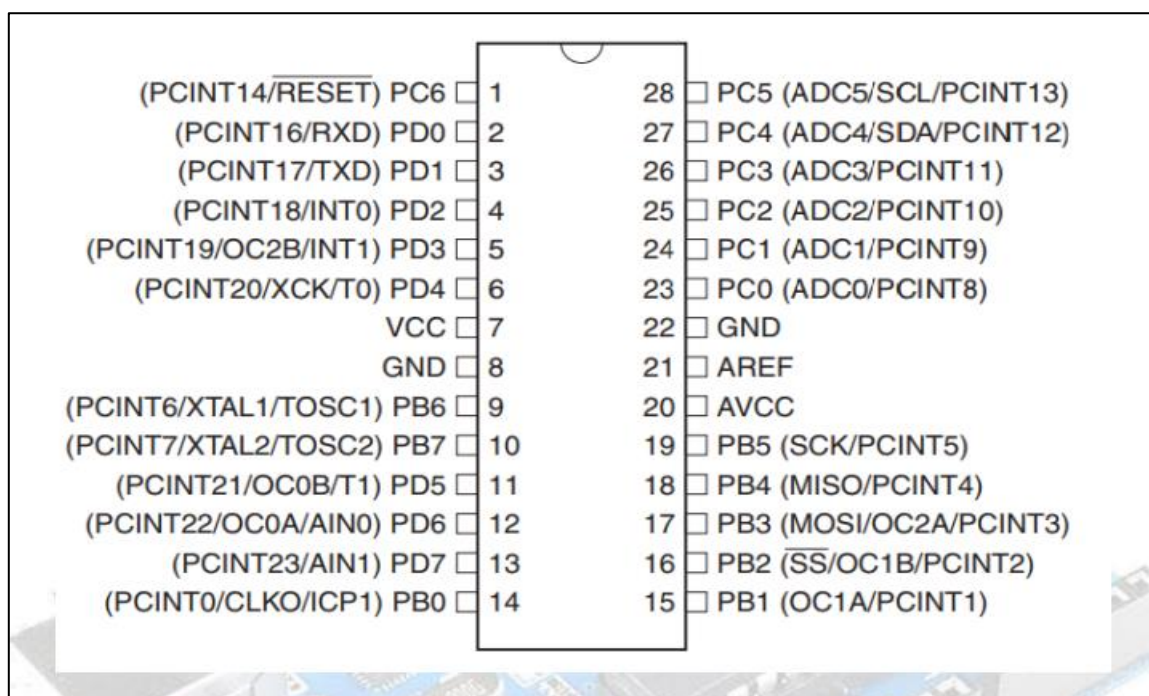
## 1.1 Arduino UNO



Hình 1.1 Arduino Uno R3

Bảng 1.1 Liệt kê các thông số cơ bản của Ardunio

Vi điều khiển	ATmega328 họ 8bit
Điện áp hoạt động	5V DC (chỉ được cấp qua cổng USB)
Tần số hoạt động	16 MHz
Dòng tiêu thụ	khoảng 30mA
Điện áp vào khuyến dùng	7-12V DC
Điện áp vào giới hạn	6-20V DC
Số chân Digital I/O	14 (6 chân hardware PWM)
Số chân Analog	6 (độ phân giải 10bit)
Dòng tối đa trên mỗi chân I/O	30 mA
Dòng ra tối đa (5V)	500 mA
Dòng ra tối đa (3.3V)	50 mA
Bộ nhớ flash	32 KB (ATmega328) với 0.5KB dùng bởi bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)



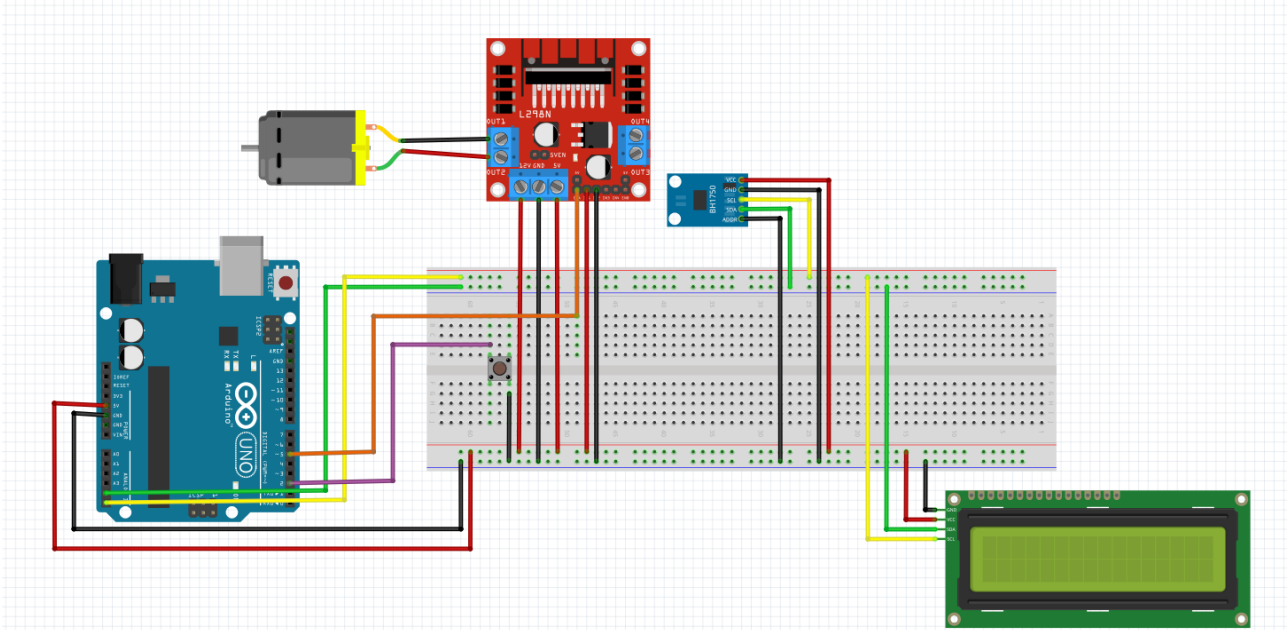
Hình 1.2 Sơ đồ chân Arduino Uno R3

## 1.2 Các linh kiện khác

Bảng 1.2 Thông tin các linh kiện khác

Linh kiện	Công dụng
Cảm biến ánh sáng BH1750	Đo cường độ ánh sáng (0 – 65535 lux)
Module cầu H L298N	Nhận tín hiệu PWM từ Arduino và điều khiển động cơ
Động cơ 6V	Xoay (27800 vòng / phút)
LCD I2C 16x2	Hiển thị cường độ ánh sáng và công suất động cơ
Button	Interrupt
Keypad	Nhập Input

### 1.3 Sơ đồ mạch



Hình 1.3 Sơ đồ mạch hoàn thiện

## PHẦN 2: Ý TƯỞNG

Gồm 3 giai đoạn:

- Giai đoạn 1: Đọc tín hiệu từ cảm biến ánh sáng
- Giai đoạn 2: PWM dựa trên tín hiệu đọc được
- Giai đoạn 3: Xuất tín hiệu lên động cơ

**Giai đoạn 1:** Đọc giá trị cảm biến ánh sáng BH1750

Thông số thực tế cảm biến đo được:

Vào buổi tối : 0.001 - 0.02 Lux

Ánh trắng : 0.02 - 0.3 lux

Trời nhiều mây trong nhà : 5 - 40 lux

Trời nhiều mây ngoài trời : 50 - 500 lux

Trời nắng: 100 - 800 lux

**Giai đoạn 2:** PWM dựa trên thông số cảm biến

0 – 40 lux: Công suất 20 %

41 – 800 lux: Công suất từ 21 – 80 %

>800 lux: Công suất 100 %

**Giai đoạn 3:** Xuất giá trị PWM sang động cơ



## PHẦN 3: CODE

### 3.1 Intro ENGINE CONTROL

```
void Intro(){
    int cntIntro = 0;
    // define variables

    int startPoint;
    int endPoint;
    int i, j;

    // speed of the text movement
    int speed = 20;

    // text to display
    String txtMsg = "ENGINE CONTROL";
    //INTRO

    startPoint = 0; //set starting point
    endPoint = 15; //set ending point
    lcd.clear();

    //for each letter of the string starting from the last one.
    for (i = txtMsg.length() - 1; i >= 0; i--)
    {
        startPoint = 0;
        //for each position on the LCD display
        for (j = 0; j < endPoint; j++)
        {

            lcd.setCursor(startPoint, 0);
            lcd.print(txtMsg[i]);

            delay(speed);

            if (startPoint != endPoint - 1) {
                lcd.setCursor(startPoint, 0);
                lcd.print(' ');
            }
            startPoint++;
        }
        endPoint--;

        delay(speed);
    }

    // hold the string on the display for 1 sec.
    delay(1000);
}
```

### 3.2 *MODE*

- MODE 0: Hiện thị bảng tùy chọn chế độ

```
if (Md == 0)
{
    analogWrite(5, 0); //Turn off engine for safety
    config();
}
```

```
void config()
{
    //display options
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("MODE: ");
    lcd.setCursor(6,0);
    lcd.print("1. AUTO");
    lcd.setCursor(6,1);
    lcd.print("2. MANUAL");
    //wait for input
    while(Serial.available()==0)
    {

    }
    MODE = Serial.parseFloat();
    //clear serial buffer
    while(Serial.available())
        Serial.read();
    lcd.clear();
    //if choose MODE 2 (MANUAL), must put 1 more 1 input - POWER
    if(MODE == 2 ){
        lcd.setCursor(0, 0);
        lcd.print("Nhap power(%): ");
        while(Serial.available()==0)
        {

        }
        valPower = Serial.parseFloat();
        while(Serial.available())
            Serial.read();
        lcd.clear();
    }
}
```

- MODE 1: Tự động dựa trên cường độ ánh sáng

```

else if(Md == 1)
{
    float lux = lightMeter.readLightLevel(); //read value light sensor
    //display
    lcd.setCursor(0, 0);
    lcd.print("LUX: ");
    lcd.setCursor(0, 1);
    lcd.print("POWER(%): ");

    //PWM
    //temp = (analog) power
    //if (lux <= 40) ~ power 20%
    if (lux <= 40)
    {
        temp = 51;
        power = 20;
    }
    //if (lux <= 800 && lux > 40) ~ power 20% -> 80%
    //increase 1 lux ~ increase(204(80%) - 51(20%)) / (800 - 40)
    else if (lux > 40 && lux <=800)
    {
        temp = ((lux - 40) * 153 / 760) + 51;
        power = ((lux - 40) * 60 / 760) + 20;
    }
    //if (lux > 800) ~ power 100%
    else
    {
        temp = 255;
        power = 100;
    }
    //dispaly read value
    analogWrite(5, temp);
    lcd.setCursor(11, 0);
    lcd.print(lux);
    lcd.setCursor(11, 1);
    lcd.print(power);
    delay(1000);
}

```

- MODE 2: Điều chỉnh công suất thủ công:

```
else if(Md == 2)
{
    int tempPow = int (valPower * 2.55);
    analogWrite(5, tempPow);
    lcd.setCursor(0, 1);
    lcd.print("POWER(%): ");
    lcd.setCursor(11, 1);
    lcd.print(valPower);
}
```

- Interrupt: Arduino hỗ trợ interrupt thông qua chân 2, được gắn bằng tín hiệu từ button. Khi nhấn nút, hệ thống sẽ quay về MODE 0, cho phép lựa chọn lại chế độ của hệ thống

```
pinMode(2, INPUT_PULLUP);
attachInterrupt(0, rst, LOW); // interrupt 0 ~ pin 2
```

```
void rst()
{
    MODE = 0;
}
```

### Source code

Link github: <https://github.com/hniman135/DoAnCuoiKi>

## **PHẦN 4: ƯU VÀ NHƯỢC ĐIỂM CỦA HỆ THỐNG**

### ***4.1 Ưu điểm***

- Cảm biến ánh sáng đọc giá trị khá chính xác
- Tốc độ phản hồi nhanh
- Hiển thị trực quan cường độ ánh sáng và công suất động cơ
- Có thể thay đổi lại chế độ

### ***4.2 Nhược điểm***

- Động cơ 6V khá yếu, nếu công suất động cơ dưới 30% thì cần tác động 1 ngoại lực nhỏ vào động cơ mới có thể xoay được
- Chức năng nhập dữ liệu còn hạn chế, phải nhập qua serial monitor do thiếu keypad
- Đóng gói chưa hoàn chỉnh