**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC NHA TRANG**



**BÁO CÁO NHÚNG**

**Môn: Lập trình thiết bị nhúng**



*GVHD: Mai Cường Thọ*

*Lớp: 61.CNTT-1*

*Tên sinh viên: Lê Quốc Duy*

*MSSV: 60135343*

MỤC LỤC

[Bài 1: Nhấp nháy Led 3](#_Toc90896012)

[Bài 2: Nút và Led 4](#_Toc90896013)

[Bài 3: Led 7 đoạn 5](#_Toc90896014)

[Bài 4: Led cảm biến nhiệt độ 6](#_Toc90896015)

[Bài 5: Led sáng dần từ 1 đến 10 8](#_Toc90896016)

[Bài 6: Led Ma Trận 9](#_Toc90896017)

[Bài 7: Đếm số từ 00 đến 99 bằng led 10](#_Toc90896018)

[Bài 8: Điều khiển độ sáng của Led thông qua chiết áp 11](#_Toc90896019)

[Bài 9: Điều khiển tín hiệu đèn giao thông 12](#_Toc90896020)

[Bài 10: Nút nhấn điều khiển các thiết bị và led 14](#_Toc90896021)

[Bài 11: Điều khiển động cơ 15](#_Toc90896022)

[Bài 12: Đọc nhiệt độ trên bảng led 17](#_Toc90896023)

[Bài 13: Đếm số từ 0 đến 9 với bảng led 7 đoạn 18](#_Toc90896024)

[Bài 14: Điều khiển sáng Led bằng Remote 22](#_Toc90896025)

[Bài 15: Tính toán với bảng tính 4x4 24](#_Toc90896026)

[Bài 16: Cảm biến khoảng cách siêu âm 27](#_Toc90896027)

[Bài 17: Xử lý ngắt với STM32 28](#_Toc90896028)

[Bài 18: Nhấp nháy lần lượt 4 led sử dụng STM32 30](#_Toc90896029)

[Bài 19: Led trái tym sử dụng STM32 33](#_Toc90896030)

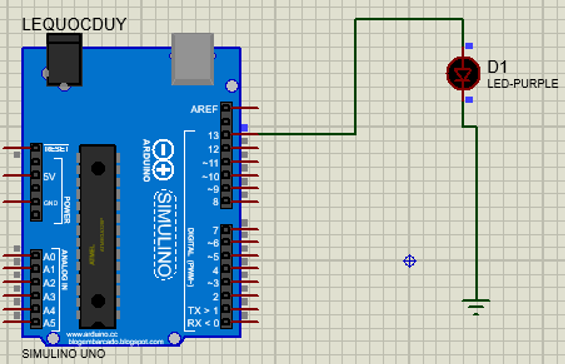
[Bài 20: Xử lý với nút bấm sử dụng STM32 42](#_Toc90896031)

# Bài 1: Nhấp nháy Led

*1.Mô tả*

Thực hiện việc nhấp nháy led liên tục. Led được bắt từ cổng 13 của Board mạch và đầu ra cắm xuống đất.

*2.Sơ đồ thiết kế*



Hình 1. Nhấp nháy Led

*3.Đặc điểm link kiện*

* Led tím: sáng led
* Board mạch: có tác dụng làm cho các thiết bị điện tử có khả năng liên kết, tương tác với nhau.

*4.Mã lệnh chính*

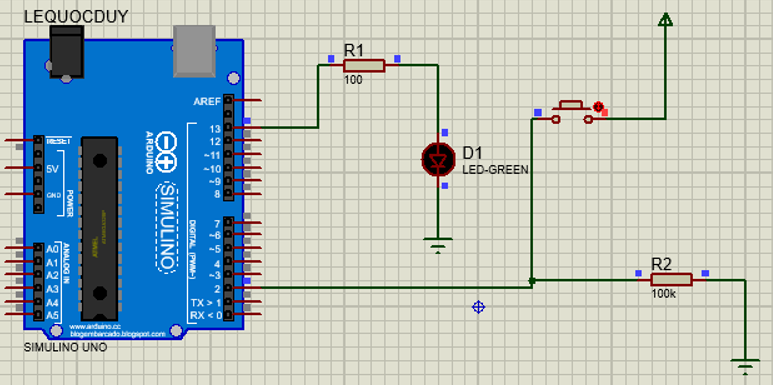
|  |
| --- |
| *int LedPin = 13;*  *void setup() {*  *pinMode(13, OUTPUT);*  *}*  *void loop() {*  *digitalWrite(LedPin, HIGH);*  *delay(1000);*  *digitalWrite(LedPin, LOW);*  *delay(1000);*  *}* |

# Bài 2: Nút và Led

*1.Mô tả*

Mô hình thực hiện việc nhấn giữ nút ở cổng số 2 thì đèn Led được gắn ở cổng 13 của Board mạch sáng liên tục, thả nút thì đèn sẽ tắt.

*2.Sơ đồ thiết kế*



Hình 2. Nhấn giữ sáng Led

*3.Đặc điểm link kiện*

* Nút bấm giữ: kích hoạt bấm giữ nút nhấn

*4.Mã lệnh chính*

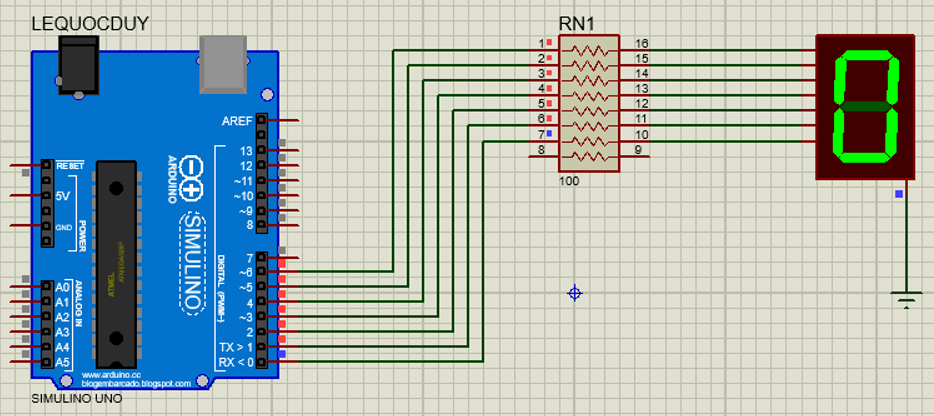
|  |
| --- |
| *int x=0;*  *int LedPin = 13;*  *int ButtonPin = 2;*  *void setup()*  *{*  *pinMode(ButtonPin, INPUT);*  *pinMode(LedPin, OUTPUT);*  *}*  *void loop() {*  *x = digitalRead(ButtonPin);*  *if (x == HIGH) {*  *digitalWrite(LedPin, HIGH);*  *} else {*  *digitalWrite(LedPin, LOW);*  *}*  *delay(1000);*  *}* |

# Bài 3: Led 7 đoạn

*1.Mô tả*

Mô hình thực hiện việc hiển thị led các số từ 0 đến 9 thông qua các chân cắm lần lượt từ 0 đến 6 của board mạch, qua thanh điện trở DIP nối tới đèn Led 7 đoạn.

*2.Sơ đồ thiết kế*



Hình 3. Hiển thị Led số 0

*3.Đặc điểm link kiện*

* Led 7 đoạn: Hiển thị các số từ 0 đến 9
* Điện trở DIP: điện trở gồm hai hàng chân cắm song song.

*4.Mã lệnh chính*

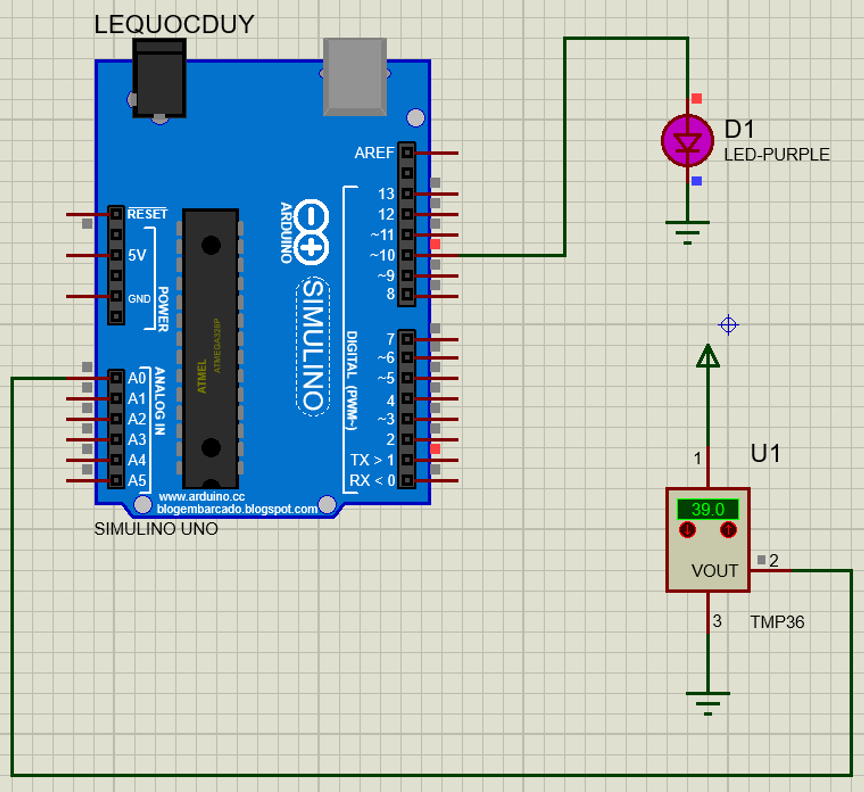
|  |
| --- |
| *int a=6,b=5,c=4,d=3,e=2,f=1,g=0;*  *void setup() {*  *pinMode(a,OUTPUT);*  *pinMode(b,OUTPUT);*  *pinMode(c,OUTPUT);*  *pinMode(d,OUTPUT);*  *pinMode(e,OUTPUT);*  *pinMode(f,OUTPUT);*  *pinMode(g,OUTPUT);*  *}*  *void KHONG(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,HIGH);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,LOW);*  *}*  *void TAM(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,HIGH);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,HIGH);*  *}*  *void loop() {*  *KHONG();*  *delay(1000);*  *TAM();*  *delay(1000);*  *}* |

# Bài 4: Led cảm biến nhiệt độ

*1.Mô tả*

Mô hình thực hiện việc sáng Led đi qua cổng số 10 của bo mạch bằng việc đọc nhiệt độ từ cảm biến, nếu nhiệt độ lớn hơn 37’C thì đèn sáng và dưới 37’C thì đèn tắt.

*2.Sơ đồ thiết kế*



Hình 4. Led cảm biến nhiệt độ

*3.Đặc điểm link kiện*

* TMP36: cảm biến nhiệt độ ( -40 -> 125 ‘C)

*4.Mã lệnh chính*

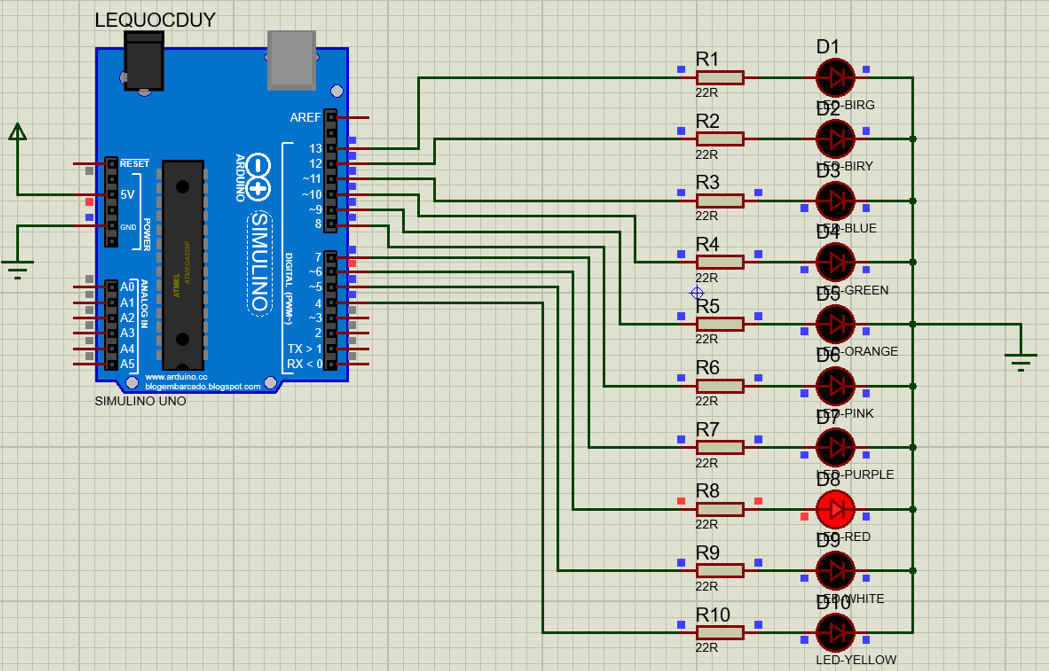
|  |
| --- |
| *void setup()*  *{*  *Serial.begin(96000);*  *pinMode(10,OUTPUT);*  *}*  *void loop()*  *{*  *int GiaTri = analogRead(A0);*  *int NhietDo=map(GiaTri,20,358,-40,125);*  *Serial.print(NhietDo);*  *Serial.print("\t");*  *if(NhietDo>37)*  *digitalWrite(10,HIGH);*  *else*  *digitalWrite(10,LOW);*  *delay(1000);*  *}* |

# Bài 5: Led sáng dần từ 1 đến 10

*1.Mô tả*

Mô hình thực hiện việc sáng Led tự động từ Led 1 đến Led 10 thông qua các cổng lần lượt từ 4 đến 13 đi qua 10 điện trở liên kết lại với nhau rồi cắm đất.

*2.Sơ đồ thiết kế*



Hình 5. Led sáng dần từ 1 đến 10

*3.Đặc điểm link kiện*

*4.Mã lệnh chính*

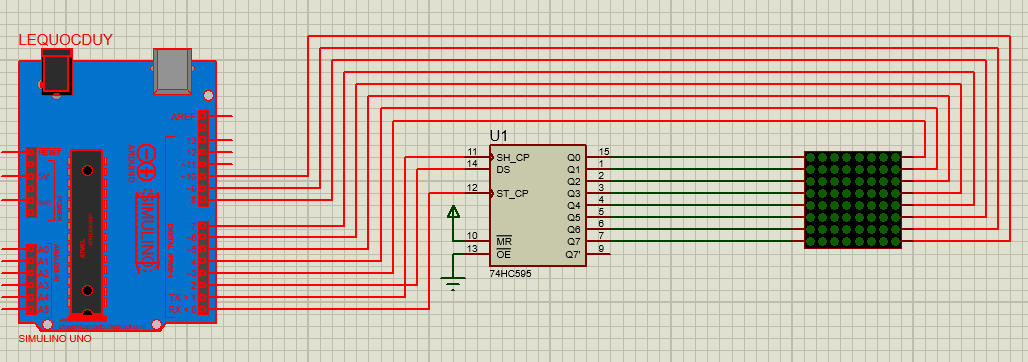
|  |
| --- |
| *byte ledPin[]={4,5,6,7,8,9,10,11,12,13};*  *int direction=1;*  *int currentLED=0;*  *void setup() {*  *for(int x=0;x<10;x++){*  *pinMode(ledPin[x],OUTPUT);*  *}*  *}*  *void loop() {*  *for(int x=0;x<10;x++){*  *digitalWrite(ledPin[x],LOW);*  *}*  *digitalWrite(ledPin[currentLED],HIGH);*  *currentLED+=direction;*  *if(currentLED==9){direction = -1;}*  *if(currentLED==0){direction = 1;}*  *delay(500);*  *}* |

# Bài 6: Led Ma Trận

*1.Mô tả*

Mô hình thực hiện việc sáng led theo ma trận, board mạch gửi dữ liệu qua 74HC595, nó nhận dữ liệu và gửi dữ liệu qua các chân song song đến Led ma trận

*2.Sơ đồ thiết kế*



Hình 6. Led ma trận

*3.Đặc điểm link kiện*

* 74HC595: thanh ghi dịch nối tiếp vào song song ra
* Đèn led Ma trận: bảng led sáng 8x8

*4.Mã lệnh chính*

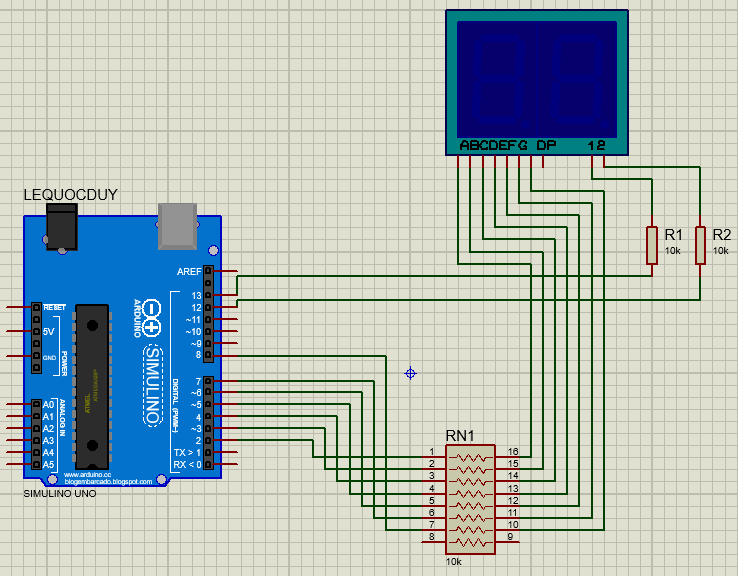
|  |
| --- |
| *#define \_clock 6*  *#define \_latch 7*  *#define \_data 5*  *void setup() {*  *pinMode(\_latch, OUTPUT);*  *pinMode(\_clock, OUTPUT);*  *pinMode(\_data, OUTPUT);*  *}*  *void loop() {*  *digitalWrite(\_latch,LOW);*  *int so = 255;*  *shiftOut(\_data,\_clock,LSBFIRST,so);*  *digitalWrite(\_latch,HIGH);*  *}* |

# Bài 7: Đếm số từ 00 đến 99 bằng led

*1.Mô tả*

Mô hình thực hiện việc đếm số từ 00 đến 99 với bảng led đôi 7 thanh cathode thông qua thanh điện trở RES16DIPIS kết nối đến với board mạch Arduino.

*2.Sơ đồ thiết kế*



Hình 7. Đếm số từ 00 đến 99

*3.Đặc điểm link kiện*

* 7SEG-MPX2-CC-BLUE: bảng hiển thị led đôi 2 số từ 00 đến 99.

*4.Mã lệnh chính*

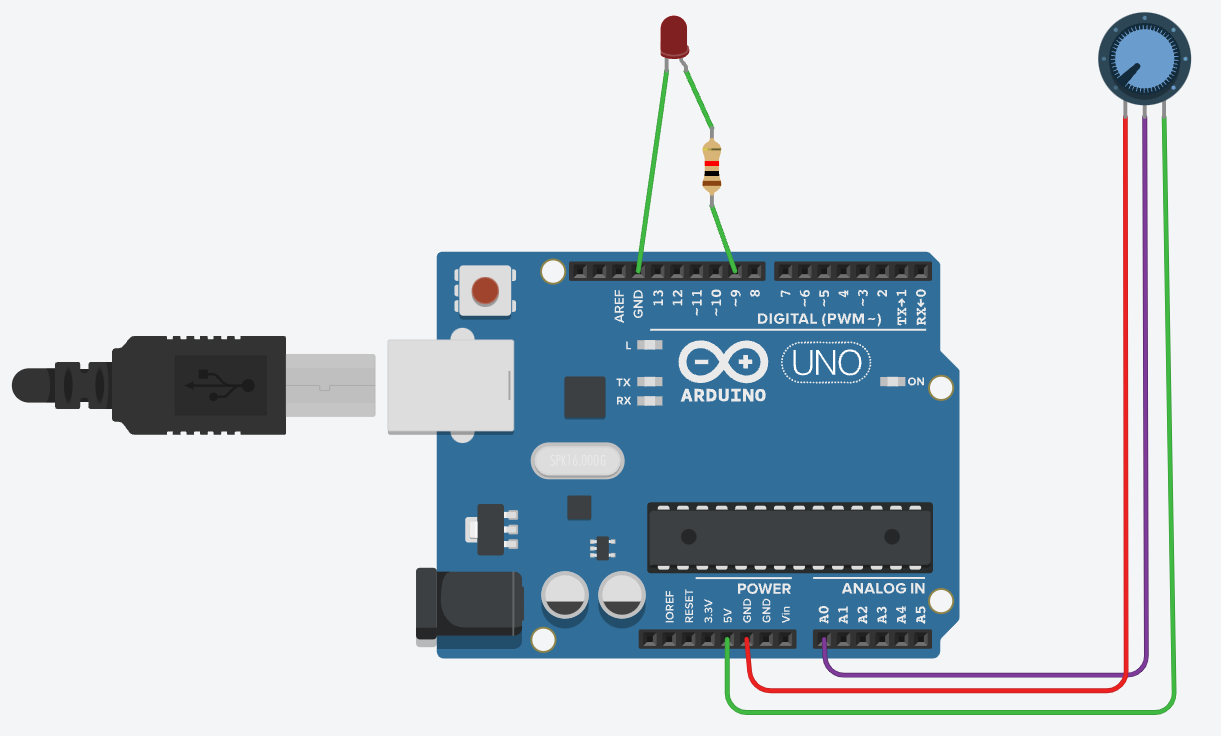
|  |
| --- |
| *int cacSo[10] = {0b0111111, 0b0000110, 0b1011011, 0b1001111, 0b1100110, 0b1101101, 0b1111101, 0b0000111, 0b1111111, 0b1101111};*  *int soThu1, soThu2;*  *void setup()*  *{*  *pinMode(2, OUTPUT);//chân a*  *pinMode(3, OUTPUT);//chân b*  *pinMode(4, OUTPUT);//chân c*  *pinMode(5, OUTPUT);//chân d*  *pinMode(6, OUTPUT);//chân e*  *pinMode(7, OUTPUT);//chân f*  *pinMode(8, OUTPUT);//chân g*  *pinMode(12, OUTPUT);//chân số 1*  *pinMode(13, OUTPUT);//chân số 2*  *}*  *void loop() {*  *for (int j = 0; j <= 99; j++) // hiện thị số từ 0 đến 99*  *{*  *soThu2 = j / 10;*  *soThu1 = j % 10;*  *for ( int k = 0; k < 20; k++)*  *{*  *digitalWrite(12, HIGH);//hiện thị số ở trước*  *digitalWrite(13, LOW);*  *ketNoi(soThu2);*  *delay(10);*  *digitalWrite(13, HIGH);//hiện thị số ở sau*  *digitalWrite(12, LOW);*  *ketNoi(soThu1);*  *delay(10);*  *}*  *}*  *}*  *void ketNoi(int num)*  *{*  *for (int i = 2; i < 9; i++) // các chân arduino cắm vào điện trở*  *{*  *digitalWrite(i, bitRead(cacSo[num], i - 2)); //bitRead trả về giá trị tại một bit i của một số nguyên tương ứng với các chân a,b,c,d,e,f,g tương ứng vơi i chạy từ 2-9 của Arduino*  *}*  *}* |

# Bài 8: Điều khiển độ sáng của Led thông qua chiết áp

*1.Mô tả*

Mô hình thực hiện việc sáng dần Led được cắm ở chân số 9 của bo mạch thông qua việc xoay nút theo chiều từ trái sang phải.

*2.Sơ đồ thiết kế*



Hình 8. Led sáng dần thông qua điều khiển chiết áp

*3.Đặc điểm link kiện*

* Chiết áp:  là một biến trở có thể điều chỉnh xoay (gồm 3 chân: chân đất, chân mắc vào đầu ra của mạch điện và một mắc vào đầu vào của mạch)

*4.Mã lệnh chính*

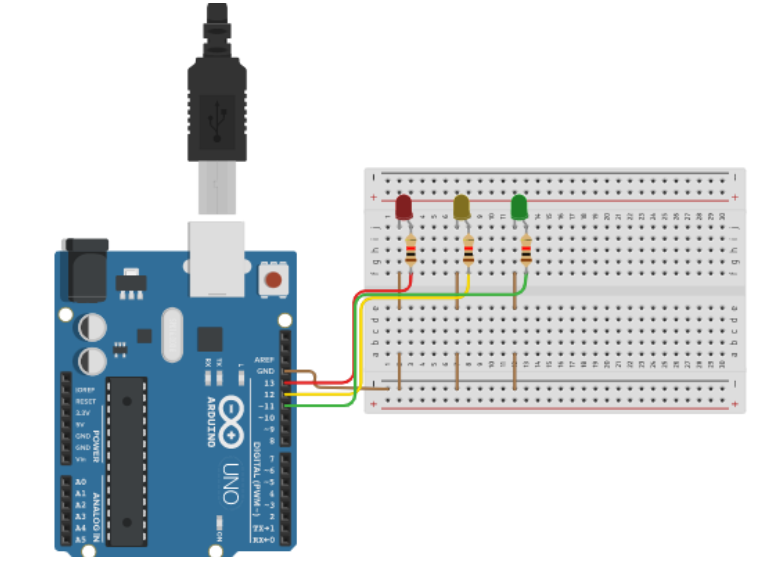
|  |
| --- |
| *int x = 0;*  *void setup()*  *{*  *pinMode(A0, INPUT);*  *pinMode(9, OUTPUT);*  *}*  *void loop()*  *{*  *x = analogRead(A0);*  *int brightness = map(x,0,1023,0,255);*  *analogWrite(9,brightness);*  *}* |

# Bài 9: Điều khiển tín hiệu đèn giao thông

*1.Mô tả*

Mô hình thực hiện việc sáng lần lượt các led đỏ chạy 15s, vàng chạy 5s, xanh chạy 25s tượng trưng cho tín hiệu đèn giao thông, được cắm vào bảng mạch đi qua các cổng 13 (đỏ) 12 (vàng) 11(xanh).

*2.Sơ đồ thiết kế*



Hình 9. Đèn led tín hiệu giao thông

*3.Đặc điểm link kiện*

* Bảng mạch: để kết nối các mạch điện mà không cần hàn.

*4.Mã lệnh chính*

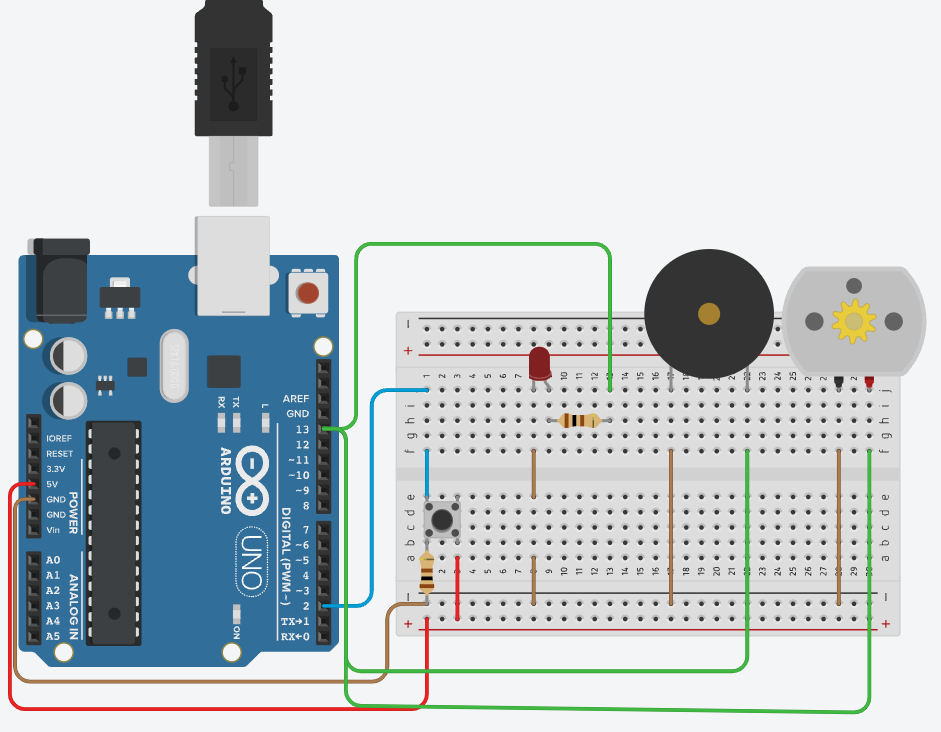
|  |
| --- |
| *#define Red 13*  *#define Yellow 12*  *#define Green 11*  *void setup()*  *{*  *pinMode(Red, OUTPUT);*  *pinMode(Yellow, OUTPUT);*  *pinMode(Green, OUTPUT);*  *}*  *void loop()*  *{*  *digitalWrite(Red, HIGH);*  *delay(15000);*  *digitalWrite(Red, LOW);*  *delay(1000);*    *digitalWrite(Yellow, HIGH);*  *delay(5000);*  *digitalWrite(Yellow, LOW);*  *delay(1000);*    *digitalWrite(Green, HIGH);*  *delay(25000);*  *digitalWrite(Green, LOW);*  *delay(1000);*  *}* |

# Bài 10: Nút nhấn điều khiển các thiết bị và led

*1.Mô tả*

Mô hình thực hiện thao tác ấn giữ nút công tắc để kích hoạt đồng thời đèn led, động cơ quay và loa, với đầu vào ở cồng số 2 của công tắc và đầu ra ở cổng 13 để điều khiển các thiết bị và đèn led.

*2.Sơ đồ thiết kế*



Hình 10. Nút bấm điều khiển Led và các thiết bị

*3.Đặc điểm link kiện*

* Loa: thiết bị điều khiển phát ra âm thanh
* Động cơ quay: thiết bị điều khiển động cơ quay

*4.Mã lệnh chính*

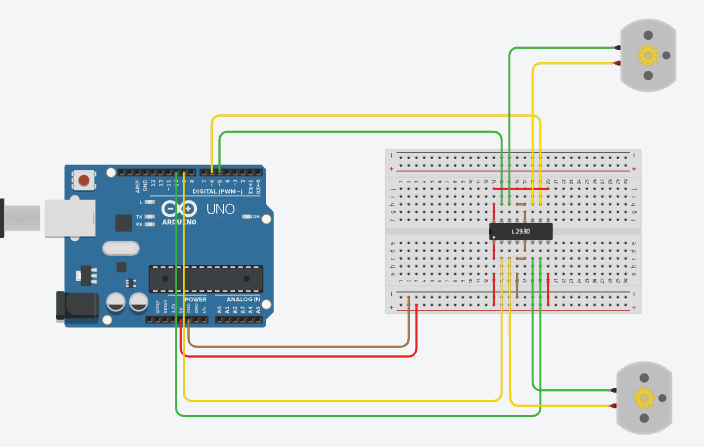
|  |
| --- |
| *#define nutBam 2*  *#define Active 13*  *void setup()*  *{*  *pinMode(Active, OUTPUT);*  *pinMode(nutBam, INPUT);*  *}*  *void loop()*  *{*  *int x = digitalRead(nutBam);*  *Serial.print(x);*  *if(x==HIGH){*  *digitalWrite(Active,HIGH);*  *}else{*  *digitalWrite(Active,LOW);*  *}*    *}* |

# Bài 11: Điều khiển động cơ

*1.Mô tả*

Mô hình thực hiện việc điều khiển một lúc 2 động cơ lần lượt về tốc độ lẫn hướng quay, được truyền dẫn thông qua một Driver Motor Shield L293D – một motor điều khiển động cơ.

*2.Sơ đồ thiết kế*



Hình 11. Điều khiển động cơ

*3.Đặc điểm link kiện*

* L293D: module mở rộng chuyên dụng cho các ứng dụng điều khiển động cơ, robot, gồm ( 2 động cơ Servo, 4 động cơ DC, 2 động cơ đơn cực hoặc lưỡng cực).

*4.Mã lệnh chính*

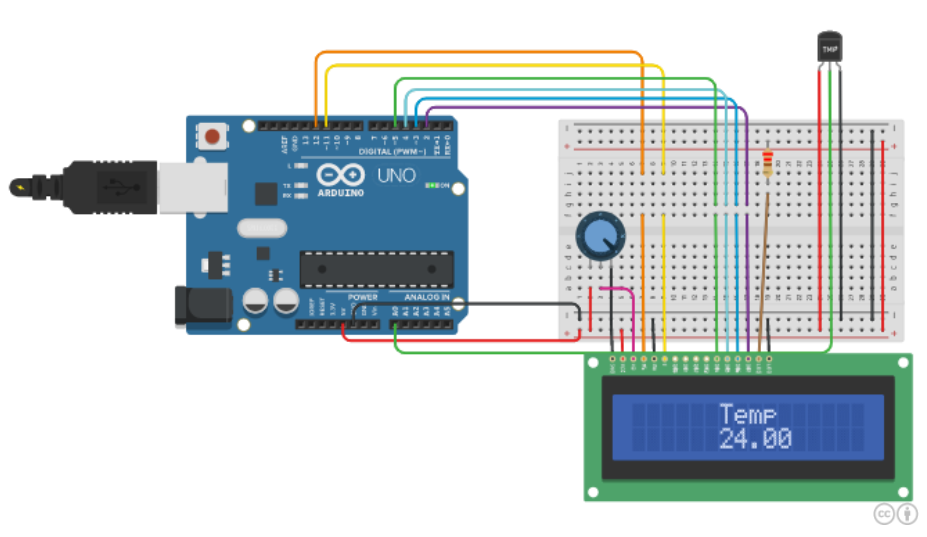
|  |
| --- |
| *//Kết nối động cơ A*  *int enA = 10;*  *int in1 = 9;*  *int in2 = 10;*  *//Kết nối động cơ B*  *int enB = 6;*  *int in3 = 5;*  *int in4 = 6;*  *void setup()*  *{*  *pinMode(enA, OUTPUT);*  *pinMode(enB, OUTPUT);*  *pinMode(in1, OUTPUT);*  *pinMode(in2, OUTPUT);*  *pinMode(in3, OUTPUT);*  *pinMode(in4, OUTPUT);*    *//Tắt cả 2 động cơ*  *digitalWrite(in1, LOW);*  *digitalWrite(in2, LOW);*  *digitalWrite(in3, LOW);*  *digitalWrite(in4, LOW);*  *}*  *void loop()*  *{*  *// directionControl();*  *// delay(1000);*  *speedControl();*  *delay(1000);*  *}*  *void directionControl(){*  *//đặt động cơ ở tốc độ tối đa*  *analogWrite(enA, 255);*  *analogWrite(enB, 255);*    *//Mở chạy động cơ A và B*  *digitalWrite(in1, HIGH);*  *digitalWrite(in2, LOW);*  *digitalWrite(in3, HIGH);*  *digitalWrite(in4, LOW);*  *delay(2000);*    *//Đổi hướng chạy động cơ*  *digitalWrite(in1, LOW);*  *digitalWrite(in2, HIGH);*  *digitalWrite(in3, LOW);*  *digitalWrite(in4, HIGH);*  *delay(2000);*    *//Tắt động cơ*  *digitalWrite(in1, LOW);*  *digitalWrite(in2, LOW);*  *digitalWrite(in3, LOW);*  *digitalWrite(in4, LOW);*  *}*  *void speedControl(){*  *//Khởi chạy 2 động cơ*  *digitalWrite(in1, LOW);*  *digitalWrite(in2, HIGH);*  *digitalWrite(in3, LOW);*  *digitalWrite(in4, HIGH);*    *//chạy động cơ từ chậm đến nhanh*  *for(int i=0; i<256; i++)*  *{*  *analogWrite(enA, i);*  *analogWrite(enB, i);*  *delay(20);*  *}*    *//Chạy động cơ từ nhanh về chậm*  *for(int i=255; i>=0; --i)*  *{*  *analogWrite(enA, i);*  *analogWrite(enB, i);*  *delay(20);*  *}*    *//Tắt động cơ*  *digitalWrite(in1, LOW);*  *digitalWrite(in2, LOW);*  *digitalWrite(in3, LOW);*  *digitalWrite(in4, LOW);*  *}* |

# Bài 12: Đọc nhiệt độ trên bảng led

*1.Mô tả*

Mô hình thực hiện việc thay đổi nhiệt độ của cảm biến sẽ hiện trực tiếp trên bảng Led.

*2.Sơ đồ thiết kế*



Hình 12. Đọc nhiệt độ với led

*3.Đặc điểm link kiện*

* Bảng Led: Hiển thị led chữ và số

*4.Mã lệnh chính*

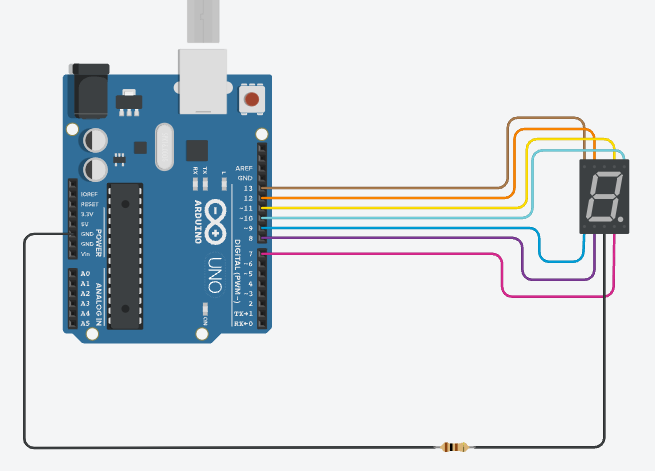
|  |
| --- |
| *#include <LiquidCrystal.h>*  *LiquidCrystal lcd(12, 11, 5, 4, 3, 2);*  *void setup() {*  *Serial.begin(96000);*  *lcd.begin(16, 2);*  *lcd.setCursor(6, 0);*  *lcd.print("Temp");*  *}*  *void loop() {*  *int GiaTri = analogRead(A0);*  *float NhietDo=map(GiaTri,20,358,-40,125);*  *lcd.setCursor(6, 1);*  *lcd.print(NhietDo);*  *}* |

# Bài 13: Đếm số từ 0 đến 9 với bảng led 7 đoạn

*1.Mô tả*

Mô hình thực hiện việc thay đổi số của bảng led 7 đoạn lần lượt từ 0 đến 9 và liên tục những lần sau đó.

*2.Sơ đồ thiết kế*



Hình 13. Đếm số từ 0 đến 9

*3.Đặc điểm link kiện*

* Bảng Led 7 đoạn (thể hiện các số 0 đến 9)

*4.Mã lệnh chính*

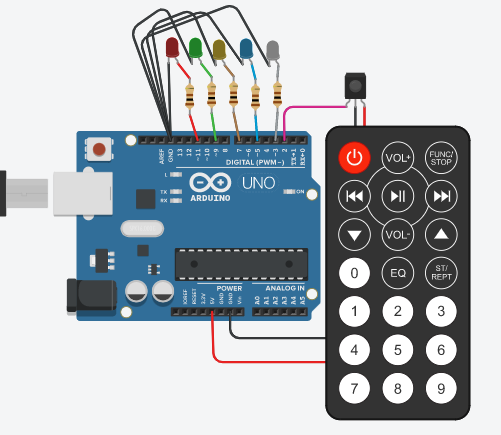
|  |
| --- |
| *int a=11,b=10,c=7,d=8,e=9,f=12,g=13;*  *void setup() {*  *pinMode(a,OUTPUT);*  *pinMode(b,OUTPUT);*  *pinMode(c,OUTPUT);*  *pinMode(d,OUTPUT);*  *pinMode(e,OUTPUT);*  *pinMode(f,OUTPUT);*  *pinMode(g,OUTPUT);*  *}*  *void KHONG(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,HIGH);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,LOW);*  *}*  *void MOT(){*  *digitalWrite(a,LOW);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,LOW);*  *digitalWrite(e,LOW);*  *digitalWrite(f,LOW);*  *digitalWrite(g,LOW);*  *}*  *void HAI(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,LOW);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,HIGH);*  *digitalWrite(f,LOW);*  *digitalWrite(g,HIGH);*  *}*  *void BA(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,LOW);*  *digitalWrite(f,LOW);*  *digitalWrite(g,HIGH);*  *}*  *void BON(){*  *digitalWrite(a,LOW);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,LOW);*  *digitalWrite(e,LOW);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,HIGH);*  *}*  *void NAM(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,LOW);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,LOW);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,HIGH);*  *}*  *void SAU(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,LOW);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,HIGH);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,HIGH);*  *}*  *void BAY(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,LOW);*  *digitalWrite(e,LOW);*  *digitalWrite(f,LOW);*  *digitalWrite(g,LOW);*  *}*  *void TAM(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,HIGH);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,HIGH);*  *}*  *void CHIN(){*  *digitalWrite(a,HIGH);*  *digitalWrite(b,HIGH);*  *digitalWrite(c,HIGH);*  *digitalWrite(d,HIGH);*  *digitalWrite(e,LOW);*  *digitalWrite(f,HIGH);*  *digitalWrite(g,HIGH);*  *}*  *void loop() {*  *KHONG();*  *delay(1000);*  *MOT();*  *delay(1000);*  *HAI();*  *delay(1000);*  *BA();*  *delay(1000);*  *BON();*  *delay(1000);*  *NAM();*  *delay(1000);*  *SAU();*  *delay(1000);*  *BAY();*  *delay(1000);*  *TAM();*  *delay(1000);*  *CHIN();*  *delay(1000);*  *}* |

# Bài 14: Điều khiển sáng Led bằng Remote

*1.Mô tả*

Thực hiện việc sáng led theo phím bấm của điều khiển từ xa. Bằng việc tích hợp lệnh sáng đèn led theo mã số tìm được của các nút trên Remote.

*2.Sơ đồ thiết kế*



Hình 14. Điều khiển sáng Led

*3.Đặc điểm link kiện*

* Remote: thiết bị diều khiển từ xa bằng việc nhấn nút với việc set các mã serial để gán vào các nút
* IR sensor: hồng ngoại cảm ứng

*4.Mã lệnh chính*

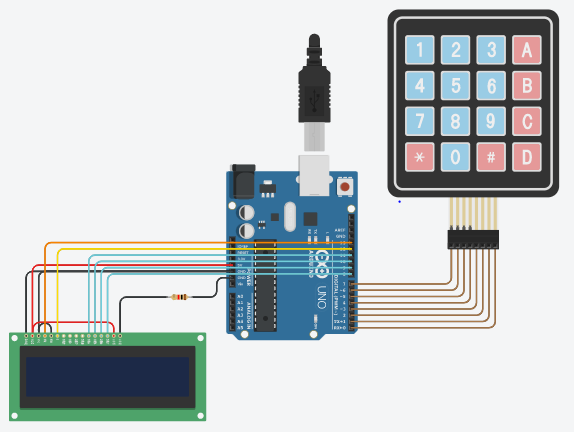
|  |
| --- |
| *#include <IRremote.h>*  *int inputDulieu = 2;*  *IRrecv boHongNgoai(inputDulieu);*  *decode\_results tinHieuThuDuoc;*  *#define Re 11*  *#define Gre 9*  *#define Ye 7*  *#define Bl 5*  *#define Gra 3*  *void setup()*  *{*  *pinMode(Re,OUTPUT);*  *pinMode(Gre,OUTPUT);*  *pinMode(Ye,OUTPUT);*  *pinMode(Gra,OUTPUT);*  *pinMode(Bl,OUTPUT);*  *pinMode(inputDulieu, INPUT);*  *Serial.begin(9600);*  *boHongNgoai.enableIRIn();*  *}*  *void loop()*  *{*  *if (boHongNgoai.decode(&tinHieuThuDuoc)){*    *if(tinHieuThuDuoc.value == 16582903) //Phím 1*  *{*  *digitalWrite(Re,HIGH);*  *}*  *if(tinHieuThuDuoc.value == 16615543) //Phím 2*  *{*  *digitalWrite(Gre,HIGH);*  *}*  *if(tinHieuThuDuoc.value == 16599223) //Phím 3*  *{*  *digitalWrite(Ye,HIGH);*  *}*  *if(tinHieuThuDuoc.value == 16591063) //Phím 4*  *{*  *digitalWrite(Bl,HIGH);*  *}*  *if(tinHieuThuDuoc.value == 16623703) //Phím 5*  *{*  *digitalWrite(Gra,HIGH);*  *}*  *if(tinHieuThuDuoc.value == 16597183) //FUNC/STOP*  *{*  *digitalWrite(Re,LOW);*  *digitalWrite(Gre,LOW);*  *digitalWrite(Ye,LOW);*  *digitalWrite(Gra,LOW);*  *digitalWrite(Bl,LOW);*  *}*  *Serial.println(tinHieuThuDuoc.value);*  *boHongNgoai.resume();*  *}*  *delay(100);*  *}* |

# Bài 15: Tính toán với bảng tính 4x4

*1.Mô tả*

Mô hình thực hiện việc tính toán các phép toán cơ bản trên bảng tính 4x4 và hiển thị trên led LCD.

*2.Sơ đồ thiết kế*



Hình 15. Thực hiện phép toán với bảng tính 4x4

*3.Đặc điểm link kiện*

* Bảng tính 4x4:dùng để tưởng tác bằng việc nhấn nút gồm 8 đầu kết nối, 4 đầu là dùng cho hàng ngang, 4 đầu dùng cho hàng dọc.

*4.Mã lệnh chính*

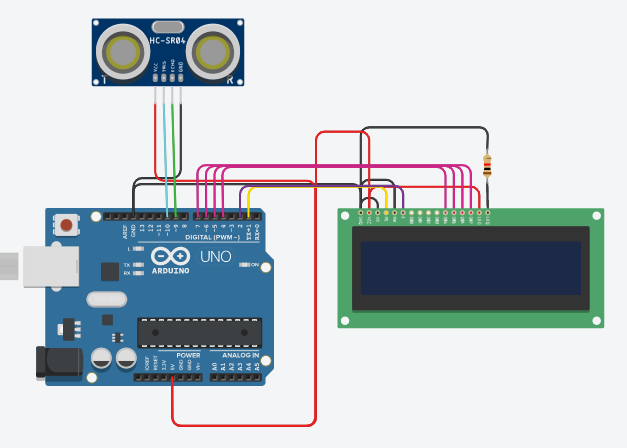
|  |
| --- |
| *#include <Keypad.h>*  *#include <Wire.h>*  *#include <LiquidCrystal.h>*  *LiquidCrystal lcd(13, 12, 11, 10, 9, 8);*  *long So1 = 0;*  *long So2 = 0;*  *double ketQua = 0;*  *char customKey;*  *const byte hang = 4;*  *const byte cot = 4;*  *char keys[hang][cot] = {*  *{'1','2','3','+'},*  *{'4','5','6','-'},*  *{'7','8','9','\*'},*  *{'C','0','=','/'}*  *};*  *byte rowPins[hang] = {7,6,5,4}; //ket noi voi so do hang bang tinh*  *byte colPins[cot] = {3,2,1,0}; //ket noi voi so do cot bang tinh*  *Keypad customKeypad = Keypad( makeKeymap(keys), rowPins, colPins, hang, cot); // khởi tạo newKeypad*  *void setup()*  *{*  *lcd.begin(16, 2);*  *for(int i=0;i<=3;i++);*  *lcd.setCursor(0,0);*  *lcd.print("Bang tinh vippro");*  *lcd.setCursor(0,1);*  *lcd.print("Copier - Duy");*  *delay(5000);*  *lcd.clear();*  *lcd.setCursor(0, 0);*  *}*  *void loop()*  *{*  *customKey = customKeypad.getKey();*  *switch(customKey)*  *{*  *// thu thập số đầu tiên*  *case '0' ... '9':*  *lcd.setCursor(0,0);*  *So1 = So1 \* 10 + (customKey - '0');*  *lcd.print(So1);*  *break;*  *//tiếp theo số đầu tiên là các toán tử*  *case '+':*  *So1 = (ketQua != 0 ? ketQua : So1);*  *lcd.setCursor(0,1);*  *lcd.print("+");*  *So2 = SecondNumber(); //sau thay thế vị trí thứ 2 vào toán tử*  *ketQua = So1 + So2;*  *lcd.setCursor(0,3);*  *lcd.print(ketQua);*  *So1 = 0, So2 = 0; // trả 2 số về 0 cho lần dùng tiếp theo*  *break;*  *case '-':*  *So1 = (ketQua != 0 ? ketQua : So1);*  *lcd.setCursor(0,1);*  *lcd.print("-");*  *So2 = SecondNumber();*  *ketQua = So1 - So2;*  *lcd.setCursor(0,3);*  *lcd.print(ketQua);*  *So1 = 0, So2 = 0;*  *break;*  *case '\*':*  *So1 = (ketQua != 0 ? ketQua : So1);*  *lcd.setCursor(0,1);*  *lcd.print("\*");*  *So2 = SecondNumber();*  *ketQua = So1 \* So2;*  *lcd.setCursor(0,3);*  *lcd.print(ketQua);*  *So1 = 0, So2 = 0;*  *break;*  *case '/':*  *So1 = (ketQua != 0 ? ketQua : So1);*  *lcd.setCursor(0,1);*  *lcd.print("/");*  *So2 = SecondNumber();*  *lcd.setCursor(0,3);*  *So2 == 0 ? lcd.print("do not use 0 !!! ") : ketQua = (float)So1 / (float)So2;*  *lcd.print(ketQua);*  *So1 = 0, So2 = 0;*  *break;*  *case 'C':*  *ketQua = 0;*  *lcd.clear();*  *break;*  *}*  *}*  *long SecondNumber()*  *{*  *while(1)*  *{*  *customKey = customKeypad.getKey();*  *if(customKey >= '0' && customKey <= '9')*  *{*  *So2 = So2 \* 10 + (customKey - '0');*  *lcd.setCursor(0,2);*  *lcd.print(So2);*  *}*  *if(customKey == '=') break; //return second;*  *}*  *return So2;*  *}* |

# Bài 16: Cảm biến khoảng cách siêu âm

*1.Mô tả*

Thực hiện việc đọc đo khoảng cách bằng sóng siêu âm rồi hiển thị trên bảng led LCD.

*2.Sơ đồ thiết kế*



Hình 16. Đo khoảng cách bằng sóng siêu âm

*3.Đặc điểm link kiện*

* HC-SR04: sử dụng sóng siêu âm đo khoảng cách trong khoảng cách từ 2 -> 300cm, gồm 4 đầu: VCC (5V), trig (chân phát), echo (chân nhận), GND

*4.Mã lệnh chính*

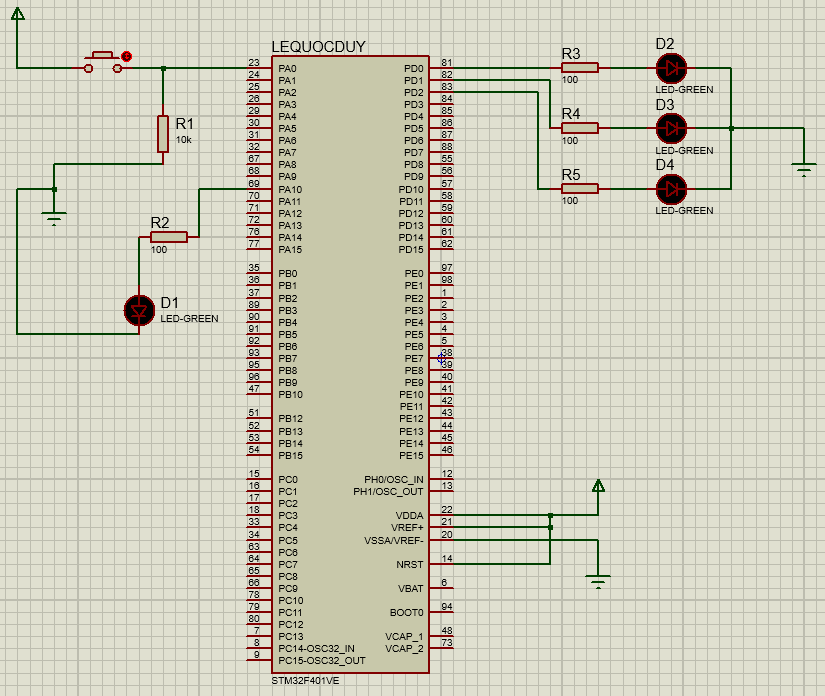
|  |
| --- |
| *#include <LiquidCrystal.h>*  *LiquidCrystal lcd(1, 2, 4, 5, 6, 7); // Tạo một đối tượng LCD. Các thông số: (rs, enable, d4, d5, d6, d7)*  *const int trigPin = 10; //chân phát tính hiệu*  *const int echoPin = 9; //chân nhận tín hiệu*  *long duration; //biến đo thời gian*  *int distanceCm, distanceInch;*  *void setup() {*  *lcd.begin(16,2);*  *pinMode(trigPin, OUTPUT);*  *pinMode(echoPin, INPUT);*  *}*  *void loop() {*  *digitalWrite(trigPin, LOW);*  *delayMicroseconds(2);*  *digitalWrite(trigPin, HIGH);*  *delayMicroseconds(10);*  *digitalWrite(trigPin, LOW);*  *duration = pulseIn(echoPin, HIGH);*  *distanceCm= duration\*0.034/2;*  *distanceInch = duration\*0.0133/2;*  *lcd.setCursor(0,0);*  *lcd.print("Distance: ");*  *lcd.print(distanceCm);*  *lcd.print(" cm");*  *delay(10);*  *lcd.setCursor(0,1);*  *lcd.print("Distance: ");*  *lcd.print(distanceInch);*  *lcd.print(" inch");*  *delay(10);*  *}* |

# Bài 17: Xử lý ngắt với STM32

*1.Mô tả*

Mô hình thực hiện xử lý ngắt các công việc của hai bên sử dụng STM32F401VE, bên kia nhấp nháy led, bên này bật tắt led với nút bấm nhưng không ảnh hưởng gì đến công việc của nhau.

*2.Sơ đồ thiết kế*



Hình 17. Xử lý ngắt với STM32

*3.Đặc điểm link kiện*

*4.Mã lệnh chính*

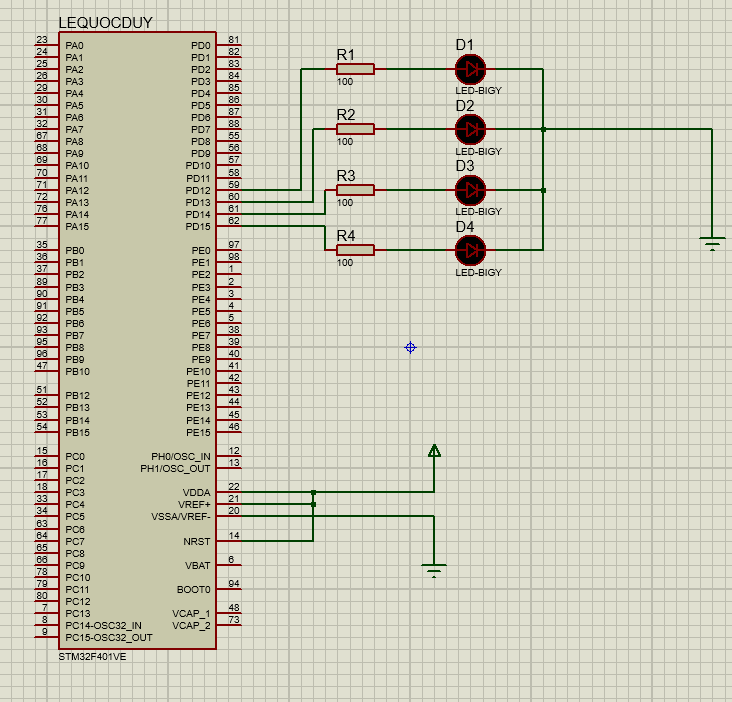
|  |
| --- |
| *#include "main.h"*  *void SystemClock\_Config(void);*  *static void MX\_GPIO\_Init(void);*  *int main(void)*  *{*    *HAL\_Init();*  *SystemClock\_Config();*  *MX\_GPIO\_Init();*  *while (1)*  *{*  *HAL\_GPIO\_TogglePin(GPIOD, GPIO\_PIN\_0|GPIO\_PIN\_1|GPIO\_PIN\_2);*  *HAL\_Delay(1000);*  *}*    *}*  *void HAL\_GPIO\_EXTI\_Callback(uint16\_t GPIO\_Pin){*  *if(GPIO\_Pin == GPIO\_PIN\_0){*  *//HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_10, GPIO\_PIN\_SET);*  *HAL\_GPIO\_TogglePin(GPIOA, GPIO\_PIN\_10);*  *}*  *}* |

# Bài 18: Nhấp nháy lần lượt 4 led sử dụng STM32

*1.Mô tả*

Mô hình thực hiện việc nhấp nháy led lần lượt và liên tục 4 led sử dụng STM32F401VE cắm vào các cổng PD từ 12 đến 15.

*2.Sơ đồ thiết kế*



Hình 18. Nháy 4 led

*3.Đặc điểm link kiện*

* STM32F401VE:  cung cấp một bộ định thời 12 bit ADC, một RTC công suất thấp, sáu bộ định thời 16 bit mục đích chung bao gồm một bộ định thời PWM để điều khiển động cơ, hai bộ định thời 32 bit mục đích chung. Chúng cũng có các giao diện giao tiếp tiêu chuẩn và nâng cao.

*4.Mã lệnh chính*

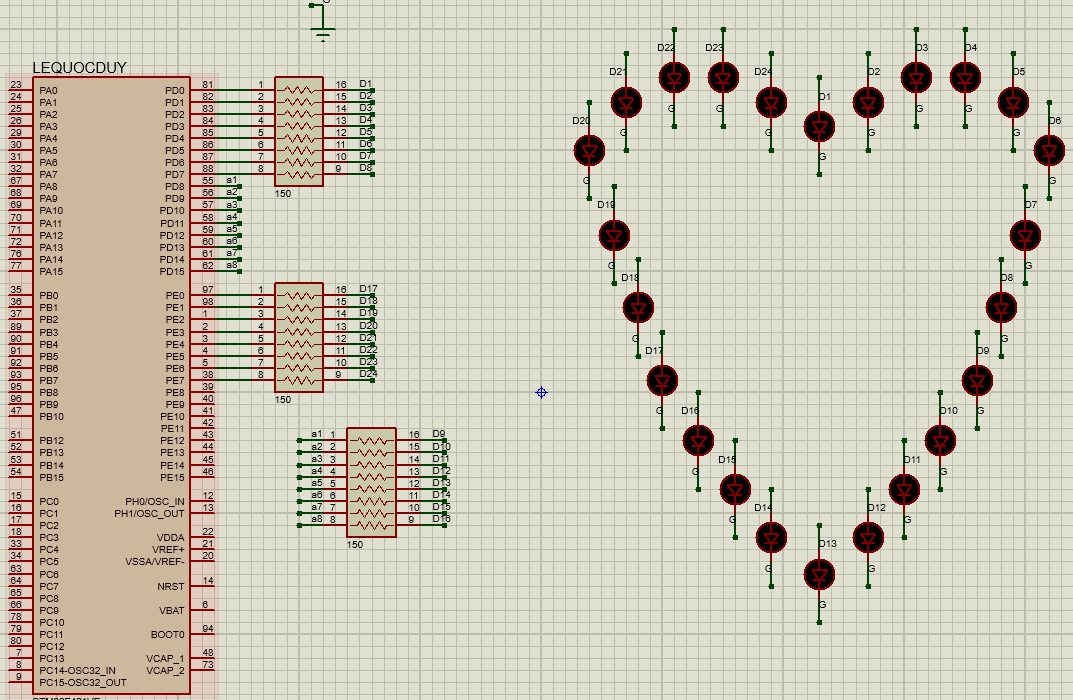
|  |
| --- |
| *#include "main.h"*  *void SystemClock\_Config(void);*  *static void MX\_GPIO\_Init(void);*  *int main(void)*  *{*  *HAL\_Init();*  *SystemClock\_Config();*  *MX\_GPIO\_Init();*    *while (1)*  *{*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_12, GPIO\_PIN\_SET);*  *HAL\_Delay(1000);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_12, GPIO\_PIN\_RESET);*    *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_13, GPIO\_PIN\_SET);*  *HAL\_Delay(1000);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_13, GPIO\_PIN\_RESET);*    *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_14, GPIO\_PIN\_SET);*  *HAL\_Delay(1000);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_14, GPIO\_PIN\_RESET);*    *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_15, GPIO\_PIN\_SET);*  *HAL\_Delay(1000);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_15, GPIO\_PIN\_RESET);*  *}*    *}* |

# Bài 19: Led trái tym sử dụng STM32

*1.Mô tả*

Mô hình thực hiện việc việc sáng led được lập trình sẵn (sáng hết led -> tắt hết led -> sáng led theo thứ tự -> sáng tắt led theo thứ tự) theo mô hình trái tym với việc sử dụng STM32F401VE và các điện trở 8 đầu vào ra - RES16DIPIS để quy định số các led tạo thành hình trái tym.

*2.Sơ đồ thiết kế*



Hình 19. Led trái tym

*3.Đặc điểm link kiện*

* STM32F401VE:  cung cấp một bộ định thời 12 bit ADC, một RTC công suất thấp, sáu bộ định thời 16 bit mục đích chung bao gồm một bộ định thời PWM để điều khiển động cơ, hai bộ định thời 32 bit mục đích chung. Chúng cũng có các giao diện giao tiếp tiêu chuẩn và nâng cao.

*4.Mã lệnh chính*

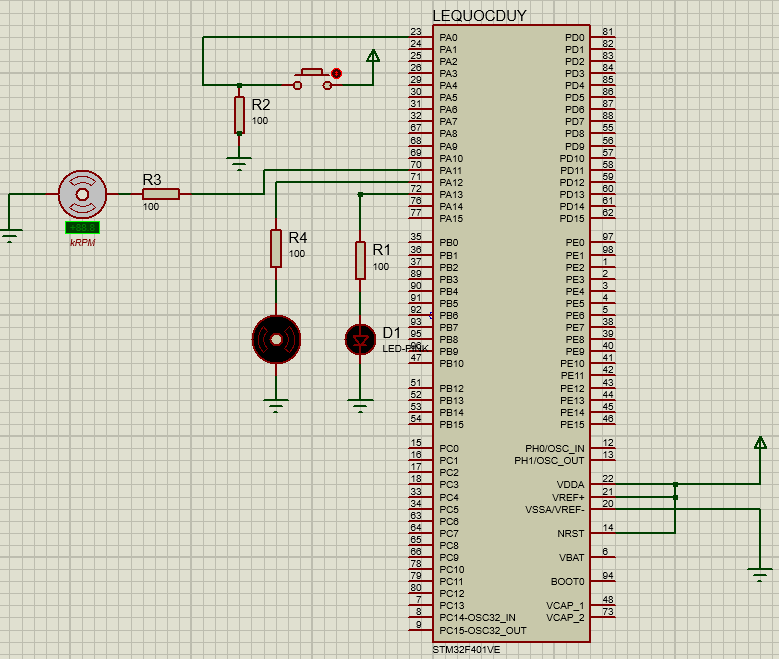
|  |
| --- |
| *#include "main.h"*  *void SystemClock\_Config(void);*  *static void MX\_GPIO\_Init(void);*  *void SangHetLed();*  *void TatHetLed();*  *void SangLanLuot();*  *void SangTatLanLuot();*  *int main(void)*  *{*  *HAL\_Init();*  *SystemClock\_Config();*  *MX\_GPIO\_Init();*    *while (1)*  *{*  *SangHetLed();*  *HAL\_Delay(1000);*  *TatHetLed();*  *HAL\_Delay(1000);*  *SangLanLuot();*  *TatHetLed();*  *HAL\_Delay(1000);*  *SangTatLanLuot();*  *TatHetLed();*  *HAL\_Delay(1000);*  *}*    *}*  *void SangHetLed(){*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_0, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_1, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_2, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_3, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_5, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_6, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_7, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_8, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_9, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_10, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_11, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_12, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_13, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_14, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_15, GPIO\_PIN\_SET);*    *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_0, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_1, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_2, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_3, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_5, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_6, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_7, GPIO\_PIN\_SET);*      *}*  *void TatHetLed(){*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_0, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_1, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_2, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_3, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_4, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_5, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_6, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_7, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_8, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_9, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_10, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_11, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_12, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_13, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_14, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_15, GPIO\_PIN\_RESET);*    *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_0, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_1, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_2, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_3, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_4, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_5, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_6, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_7, GPIO\_PIN\_RESET);*  *}*  *void SangLanLuot(){*  *TatHetLed();*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_0, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_1, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_2, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_3, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_5, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_6, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_7, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_8, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_9, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_10, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_11, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_12, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_13, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_14, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_15, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_0, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_1, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_2, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_3, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_5, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_6, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_7, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *}*  *void SangTatLanLuot(){*  *TatHetLed();*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_0, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_0, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_1, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_1, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_2, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_2, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_3, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_3, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_4, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_5, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_5, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_6, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_6, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_7, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_7, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_8, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_8, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_9, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_9, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_10, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_10, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_11, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_11, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_12, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_12, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_13, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_13, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_14, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_14, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_15, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOD , GPIO\_PIN\_15, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_0, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_0, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_1, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_1, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_2, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_2, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_3, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_3, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_4, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_5, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_5, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_6, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_6, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_7, GPIO\_PIN\_SET);*  *HAL\_Delay(100);*  *HAL\_GPIO\_WritePin (GPIOE , GPIO\_PIN\_7, GPIO\_PIN\_RESET);*  *}* |

# Bài 20: Xử lý với nút bấm sử dụng STM32

*1.Mô tả*

Mô hình thực hiện việc xử lý với nút bấm trên STM32F401VE để kích hoạt các thiết bị như đèn led, xoay động cơ quạt, chạy motor.

*2.Sơ đồ thiết kế*



Hình 20. Nút bấm với STM32

*3.Đặc điểm link kiện*

* Fan-DC: thiết bị quạt
* Motor: động cơ

*4.Mã lệnh chính*

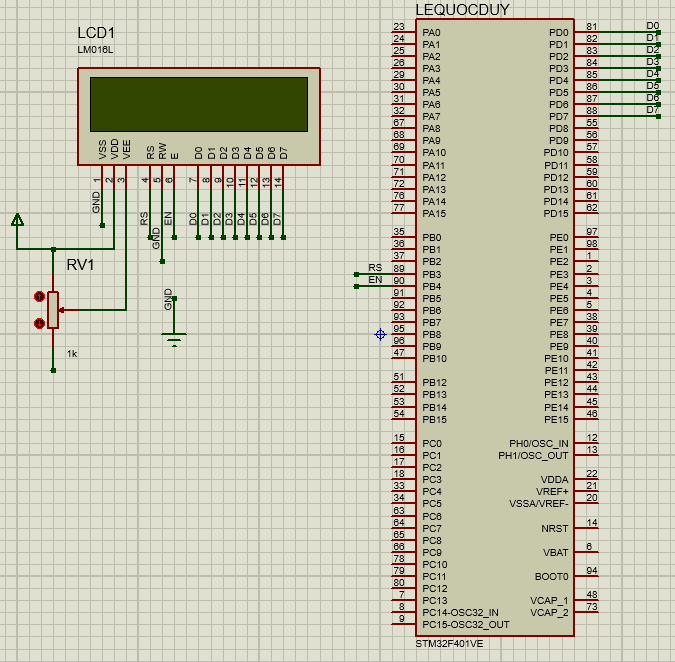
|  |
| --- |
| *#include "main.h"*  *void SystemClock\_Config(void);*  *static void MX\_GPIO\_Init(void);*  *int main(void)*  *{*    *HAL\_Init();*  *SystemClock\_Config();*  *MX\_GPIO\_Init();*  *while (1)*  *{*  *GPIO\_PinState pin0State = HAL\_GPIO\_ReadPin(GPIOA, GPIO\_PIN\_0);*  *if(pin0State == GPIO\_PIN\_SET){*  *HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_13, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_12, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_11, GPIO\_PIN\_SET);*  *}*  *else{*  *HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_13, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_12, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOA, GPIO\_PIN\_11, GPIO\_PIN\_RESET);*  *}*  *}*    *}* |

# Bài 21: Thao tác với led LCD sử dụng STM32

*1.Mô tả*

Mô hình thực hiện việc thao tác với bảng hiển thị led trên STM32F401VE về số, kí tự, chuỗi …

*2.Sơ đồ thiết kế*



Hình 21. LCD với STM32

*3.Đặc điểm link kiện*

*4.Mã lệnh chính*

|  |
| --- |
| *#include "main.h"*  *void SystemClock\_Config(void);*  *static void MX\_GPIO\_Init(void);*  *//Ham gui data/cmd ra 8 chan cua LCD*  *void send8BitLCD(char D){*  *//dem D and so hoc voi 2^i de biet bit thu i=0 hay !=0*  *//tim gia tri cac bit*  *int b0,b1,b2,b3,b4,b5,b6,b7;*  *if((D&1)==0)*  *b0=0; //1==2^0*  *else b0 = 1;*    *if((D&2)==0)*  *b1=0; //2==2^1*  *else b1 = 1;*    *if((D&4)==0)*  *b2=0; //4==2^2*  *else b2 = 1;*    *if((D&8)==0)*  *b3=0; //8==2^3*  *else b3 = 1;*    *if((D&16)==0)*  *b4=0; //16==2^4*  *else b4 = 1;*    *if((D&32)==0)*  *b5=0; //32==2^5*  *else b5 = 1;*    *if((D&64)==0)*  *b6=0; //64==2^6*  *else b6 = 1;*    *if((D&128)==0)*  *b7=0; //128==2^7*  *else b7 = 1;*    *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_0,b0);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_1,b1);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_2,b2);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_3,b3);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_4,b4);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_5,b5);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_6,b6);*  *HAL\_GPIO\_WritePin(GPIOD, GPIO\_PIN\_7,b7);*    *}*  *//Ham gui lenh*  *void sendCMD2LCD(char cmd){*  *//B1. xong*  *//B2. Dat chan RS=0, de noi rang cmd la lenh*  *HAL\_GPIO\_WritePin(GPIOB, GPIO\_PIN\_3, GPIO\_PIN\_RESET);*  *//B3. Gui 8 bit CMD vao 8 pin*  *send8BitLCD(cmd);*  *//B4. Kich hoat cmd-->lcd*  *HAL\_GPIO\_WritePin(GPIOB, GPIO\_PIN\_4, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOB, GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_Delay(1);*  *}*  *//Ham ky tu*  *void sendChar2LCD(char Char){*  *//B1. xong*  *//B2. Dat chan RS=1,*  *HAL\_GPIO\_WritePin(GPIOB, GPIO\_PIN\_3, GPIO\_PIN\_RESET);*  *//B3. Gui 8 bit CMD vao 8 pin*  *send8BitLCD(Char);*  *//B4. Kich hoat cmd-->lcd*  *HAL\_GPIO\_WritePin(GPIOB, GPIO\_PIN\_4, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOB, GPIO\_PIN\_4, GPIO\_PIN\_SET);*  *HAL\_Delay(1);*  *}*  *void sendString2LCD(char \*str){*  *for(int i=0; str[i] !='\0'; i++){*  *sendChar2LCD(str[i]);*  *}*  *}*  *int main(void)*  *{*  *HAL\_Init();*  *SystemClock\_Config();*  *MX\_GPIO\_Init();*    *//xoa noi dung co san tren LCD*  *sendCMD2LCD(0x01);*  *//Bat hien thi man hinh, tat con tro*  *sendCMD2LCD(0x0C);*  *//Viet chuoi*  *sendString2LCD("Thi LTN");*  *//xuong dong*  *sendCMD2LCD(0x38); //bat che do 2 dong*  *sendCMD2LCD(0xC0); //xuong dong 2*  *sendString2LCD("Le Quoc Duy");*    *while (1)*  *{*    *}*    *}* |