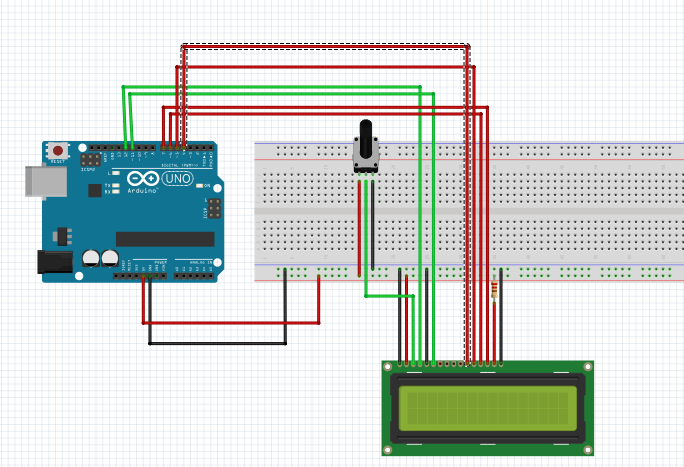
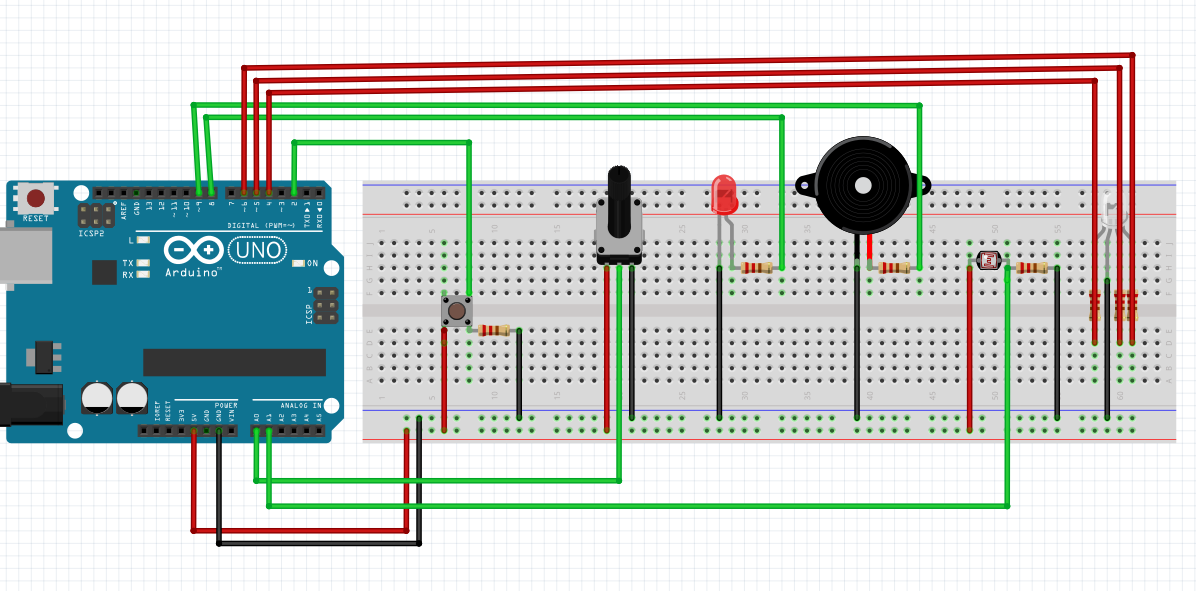
****

****

**Câu 1.** Cho thiết bị Arduino UNO và 1 biến trở được kết nối với Arduino trong đó chân số 2 của biến trở được nối vào chân analog (A0, A1,...), 1 LED màu xanh và 1 LED màu đỏ. Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau: Khi vặn biến trở, Arduino sẽ đọc giá trị điện áp tại chân analog và chuyển đổi thành nhiệt độ theo công thức sau:

**Nhiệt độ = map(Điện áp, 0,1023,0,100);**

Nếu nhiệt độ dưới 30°C, bật LED xanh

Nếu nhiệt độ trên 70°C, bật LED đỏ

Hiển thị nhiệt độ lên cửa sổ Serial Monitor cùng với thông báo đèn LED nào đang sáng.

|  |
| --- |
| const int bienTro = A0;  const int ledXanh = 8;  const int ledDo = 9;  void setup()  {  Serial.begin(9600);  pinMode(ledXanh, OUTPUT);  pinMode(ledDo, LOW);  }  void loop()  {    int giatriAnalog = analogRead(bienTro);    int nhietDo = map(giatriAnalog, 0, 1023, 0,100);    digitalWrite(ledXanh, LOW);  digitalWrite(ledDo, LOW);    if(nhietDo < 30) {  digitalWrite(ledXanh, HIGH);  Serial.print("Nhiet do: ");  Serial.print(nhietDo);  Serial.println("C - Den Xanh dang sang");  }  else if(nhietDo > 70) {  digitalWrite(ledDo, HIGH);  Serial.print("Nhiet do: ");  Serial.print(nhietDo);  Serial.println(" C - Den DO dang sang");  } else {  Serial.print("Nhiet do: ");  Serial.print(nhietDo);  Serial.println(" C - Khong co den nao sang");  }    delay(500);  } |

**Câu 2.** Cho thiết bị Arduino UNO, thực hiện kết nối 1 biến trở và 2 đèn led đơn với Arduino, trong đó chân số 2 của biến trở được nối vào một chân analog (A0, A1,...). Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau: Khi vặn biến trở, Arduino sẽ đọc giá trị điện áp tại chân analog và chuyển đổi thành nhiệt độ, điều khiển 2 đèn led đơn theo giá trị nhiệt độ như sau:

* Nếu Nhiệt độ <=30: Tất cả 2 đèn
* Nếu 31<Nhiệt độ <=50: đèn 1 sáng
* Nếu 51<Nhiệt độ <=80: đèn 2 sáng
* Nếu Nhiệt độ >80: Cả 2 đèn đều sáng

Hiển thị lên cửa sổ Serial monitor nhiệt độ và trạng thái của các đèn tương ứng khi vặn biển trở

const int bienTro = A0;

const int den1 = 8;

const int den2 = 9;

void setup() {

Serial.begin(9600);

pinMode(den1, OUTPUT);

pinMode(den2, OUTPUT);

}

void loop() {

int giatriAnalog = analogRead(bienTro);

int nhietDo = map(giatriAnalog, 0, 1023, 0, 100);

digitalWrite(den1, LOW);

digitalWrite(den2, LOW);

if (nhietDo <= 30) {

digitalWrite(den1, HIGH);

digitalWrite(den2, HIGH);

Serial.print("Nhiet do: "); Serial.print(nhietDo); Serial.println(" C - Ca 2 den dang sang");

} else if (nhietDo > 30 && nhietDo <= 50) {

digitalWrite(den1, HIGH);

Serial.print("Nhiet do: "); Serial.print(nhietDo); Serial.println(" C - Chi LED1 sang");

} else if (nhietDo > 50 && nhietDo <= 80) {

digitalWrite(den2, HIGH);

Serial.print("Nhiet do: "); Serial.print(nhietDo); Serial.println(" C - Chi LED2 sang");

} else if (nhietDo > 80) {

digitalWrite(den1, HIGH);

digitalWrite(den2, HIGH);

Serial.print("Nhiet do: "); Serial.print(nhietDo); Serial.println(" C - Ca 2 den dang sang");

}

delay(500);

}

Câu 3. Cho thiết bị Arduino UNO, thực hiện kết nối 1 biến trở và 1 đèn LED đơn với Arduino (1 chân của LED đơn được kết nối vào 1 chân PWM trên Arduino, chân số 2 của biến trở được nổi vào chân analog A0 trên Arduino). Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau: Khi xoay biển trở, độ sáng của đèn LED sẽ thay đổi tương ứng. Đồng thời, hiển thị giá trị của biển trở và độ sáng của đèn LED lên cửa sổ Serial Monitor.

|  |
| --- |
| const int bienTro = A0;  const int denLED\_PMW = 9;  void setup() {  Serial.begin(9600);  pinMode(denLED\_PMW, OUTPUT);  }  void loop() {  int giatriAnalog = analogRead(bienTro);  int dosang = map(giatriAnalog, 0, 1023, 0, 100);  analogWrite(denLED\_PMW, dosang);  Serial.print("Gia tri bien tro la: ");  Serial.print(giatriAnalog);  Serial.print(" | Do sang cua LED la: ");  Serial.print(dosang);  delay(500);  } |

**Câu 4.** Cho thiết bị Arduino UNO, thực hiện kết nối 1 biến trở và 1 còi (buzzer) với Arduino, trong đó chân số 2 của biến trở được nối vào chân analog A1, một chân của còi được nối với 1 chân PWM trên Arduino. Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau:

Arduino sẽ đọc giá trị điện áp tại chân analog A0 khi xoay biến trở và chuyển đổi thành độ to nhỏ của còi theo công thức sau:

**Âm lượng còi = map(Điện áp, 0,1023,0,255);**

Hãy hiển thị giá trị của điện áp tại chân analog Al khi xoay biến trở và âm lượng của còi tương ứng khi xoay biến trở lên cửa số Serial Monitor.

|  |
| --- |
| const int bienTro = A0;  const int denLED\_PMW = 9;  void setup() {  Serial.begin(9600);  pinMode(denLED\_PMW, OUTPUT);  }  void loop() {  int giatriAnalog = analogRead(bienTro);  int dosang = map(giatriAnalog, 0, 1023, 0, 100);  analogWrite(denLED\_PMW, dosang);  Serial.print("Gia tri bien tro la: ");  Serial.print(giatriAnalog);  Serial.print(" | Do sang cua LED la: ");  Serial.print(dosang);  delay(500);  } |

Câu 5. Cho thiết bị Arduino UNO, thực hiện kết nối 1 nút nhấn và 1 còi (buzzer) với Arduino. Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau: Khi nhấn và giữ nút nhấn còi sẽ kêu, khi thả nút nhấn ra còi sẽ ngừng kêu. Đồng thời, hiển thị lên cửa sổ Serial Monitor thông báo tương ứng: “ Nút nhấn đang được nhấn (hoặc không được nhấn), còi đang kêu (hoặc còi ngừng kêu)"

|  |
| --- |
| const int nutbam = 2;  const int coi = 9;  void setup() {  Serial.begin(9600);  pinMode(nutbam, INPUT\_PULLUP);  pinMode(coi, OUTPUT);  }  void loop() {  int trangthaiNut = digitalRead(nutbam);  if (trangthaiNut == LOW) {  digitalWrite(coi, HIGH);  Serial.println("Dang nhan nut - Coi keu");  } else {  digitalWrite(coi, LOW);  Serial.println("kHONG nhan nut - Coi tat");  }  delay(500);  } |

Câu 6.

Câu 7. Cho thiết bị Arduino UNO. Cho động cơ Servo có chân điều khiển nối vào chân 6. Cho còi Buzzer có chân điều khiển nối vào chân 7. Trong chu kỳ 7 giây, thiết bị sinh một số ngẫu nhiên a từ 0 đến 180. Động cơ sẽ trỏ tới vị trí góc a. Còi sẽ kêu với giá trị a x 200.

|  |
| --- |
| #include <Servo.h>  Servo myServo;  int buzzerPin = 7;  void setup() {  myServo.attach(6);  pinMode(buzzerPin, OUTPUT);  randomSeed(analogRead(0));  }  void loop() {  int a = random(0, 181); // random 0-180  myServo.write(a);  tone(buzzerPin, a \* 200); // tần số âm thanh theo góc  delay(500); // cho kêu 0.5 giây  noTone(buzzerPin);    delay(7000); // đợi 7 giây  } |

**Câu 8.** Cho thiết bị Arduino UNO. Cho một nút bấm nối vào chân 6. Cho một thiết bị LED 7 đoạn. Thí sinh tự thiết kế kết nối các chân của LED 7 đoạn với các chân của thiết bị (trừ chân số 6 nối vào nút bấm) để đảm bảo LED 7 đoạn hoạt động được. Khối LED sẽ hiển thị số a được khởi tạo là 0. Khi người dùng bấm nút và nhả ra thì a sẽ tăng lên 1 nếu giá trị a hiện tại lớn hơn 9 hoặc a sẽ bằng 0 nếu giá trị a hiện tại bằng 9. Khối LED 7 đoạn sẽ cập nhập giá trị a.

|  |
| --- |
| const int buttonPin = 6;  int ledPins[7] = {2, 3, 4, 5, 7, 8, 9}; // các chân nối đến a-g  int digits[10][7] = {  {1,1,1,1,1,1,0}, // 0  {0,1,1,0,0,0,0}, // 1  {1,1,0,1,1,0,1}, // 2  {1,1,1,1,0,0,1}, // 3  {0,1,1,0,0,1,1}, // 4  {1,0,1,1,0,1,1}, // 5  {1,0,1,1,1,1,1}, // 6  {1,1,1,0,0,0,0}, // 7  {1,1,1,1,1,1,1}, // 8  {1,1,1,0,0,1,1} // 9  };  int a = 0;  bool lastButtonState = LOW;  void setup() {  for (int i = 0; i < 7; i++) pinMode(ledPins[i], OUTPUT);  pinMode(buttonPin, INPUT\_PULLUP); // dùng trở kéo lên  showDigit(a);  }  void loop() {  bool buttonState = digitalRead(buttonPin) == LOW;    if (buttonState && !lastButtonState) {  a = (a == 9) ? 0 : a + 1;  showDigit(a);  delay(200); // chống dội nút  }    lastButtonState = buttonState;  }  void showDigit(int num) {  for (int i = 0; i < 7; i++) {  digitalWrite(ledPins[i], digits[num][i]);  }  } |

**Câu 9.** Cho thiết bị Arduino UNO. Sử dụng một cảm biến nhiệt độ (như LM35) có chân tín hiệu nối vào chân A1. Sử dụng một còi Buzzer có chân điều khiển nối vào chân số 6. Thiết bị sẽ đọc giá trị nhiệt độ và gán vào biến t (đơn vị: độ C). Sau đó, còi sẽ phát âm thanh với tần số f, trong đó f = t \* 50. Nếu nhiệt độ thấp (t < 15°C) hoặc cao bất thường (t > 47°C), còi sẽ không phát âm.

|  |
| --- |
| const int chanLDR = A2;  const int chanBuzzer = 10;  void setup() {  pinMode(chanBuzzer, OUTPUT);  }  void loop() {  int giaTri = analogRead(chanLDR); // Đọc giá trị ánh sáng (0–1023)  if (giaTri == 0) {  tone(chanBuzzer, 10000); // Kêu với tần số 10000 Hz  } else {  noTone(chanBuzzer); // Không kêu  }  delay(1000); // Đợi 1 giây  } |

**Câu 10.** Cho thiết bị Arduino UNO và một LED 7 đoạn. Sinh viên tự thiết kế kết nối các chân của LED 7 đoạn. Thiết kế mạch sao cho LED 7 đoạn hiển thị các số từ 0 đến 9 tuần tự trong chu kỳ 5 giây. Mỗi số sẽ được hiển thị trong khoảng 0.5 giây, bắt đầu từ số 0 đến số 9 rồi lặp lại chu kỳ này.

|  |
| --- |
| // Gán các chân LED 7 đoạn bằng mảng (a→g)  const int ledPins[7] = {2, 3, 4, 5, 6, 7, 8}; // a, b, c, d, e, f, g  // Bảng mã hiển thị số 0–9 (Common Cathode: HIGH = sáng)  const byte so[10][7] = {  // a b c d e f g  {1, 1, 1, 1, 1, 1, 0}, // 0  {0, 1, 1, 0, 0, 0, 0}, // 1  {1, 1, 0, 1, 1, 0, 1}, // 2  {1, 1, 1, 1, 0, 0, 1}, // 3  {0, 1, 1, 0, 0, 1, 1}, // 4  {1, 0, 1, 1, 0, 1, 1}, // 5  {1, 0, 1, 1, 1, 1, 1}, // 6  {1, 1, 1, 0, 0, 0, 0}, // 7  {1, 1, 1, 1, 1, 1, 1}, // 8  {1, 1, 1, 1, 0, 1, 1} // 9  };  void setup() {  // Khai báo các chân là OUTPUT  for (int i = 0; i < 7; i++) {  pinMode(ledPins[i], OUTPUT);  }  }  void hienThiSo(int n) {  for (int i = 0; i < 7; i++) {  digitalWrite(ledPins[i], so[n][i]);  }  }  void loop() {  for (int i = 0; i <= 9; i++) {  hienThiSo(i);  delay(500); // Hiển thị 0.5 giây  }  } |

**Câu 11.** Thiết kế và lập trình một hệ thống điều khiển 5 đèn LED đơn sử dụng vi điều khiển Arduino. Hệ thống thực hiện các hiệu ứng ánh sáng theo yêu cầu sau (Các hiệu ứng lặp lại liên tục):

a. Nhấp nháy toàn bộ

• Tất cả 5 LED cùng bật sáng trong một khoảng thời gian cố định, sau đó tắt toàn bộ cùng lúc.

b. Chạy sáng từ trái sang phải

• LED sáng lần lượt từ LED 1 đến LED 5.

• Mỗi LED sáng một khoảng thời gian cho trước trước khi chuyển sang sáng LED tiếp theo.

c. Tắt dần từ phải sang trái

• LED tắt lần lượt từ LED 5 đến LED 1.

• Mỗi LED tắt một khoảng thời gian cho trước trước khi chuyển sang tắt LED tiếp theo.

Tất cả các chân GND của LED nối vào GND Arduino thông qua các điện trở.

|  |
| --- |
| const int ledPins[5] = {2, 3, 4, 5, 6};  void setup() {  for (int i = 0; i < 5; i++) {  pinMode(ledPins[i], OUTPUT);  }  }  void nhapNhanhToanBo() {  for (int i = 0; i < 5; i++) {  digitalWrite(ledPins[i], HIGH);  }  delay(500);  for (int i = 0; i < 5; i++) {  digitalWrite(ledPins[i], LOW);  }  delay(500);  }  void chaySangTuTraiSangPhai() {  for (int i = 0; i < 5; i++) {  digitalWrite(ledPins[i], HIGH);  delay(500);  digitalWrite(ledPins[i], LOW);  }  }  void tatDanTuPhaiSangTrai() {  for (int i = 4; i >= 0; i--) {  digitalWrite(ledPins[i], LOW);  delay(500);  }  } |

**Câu 12.** Hệ thống đèn giao thông gồm 3 đèn LED mô phỏng ba tín hiệu:

• Đèn đỏ (Dừng)

• Đèn vàng (Chuẩn bị)

• Đèn xanh (Đi)

Viết chương trình sử dụng vi điều khiển Arduino điều khiển các đèn LED theo trình tự như sau:

1. Đèn đỏ sáng trong 3 giây (các đèn khác tắt).

2. Đèn vàng sáng trong 1 giây (các đèn khác tắt).

3. Đèn xanh sáng trong 7 giây (các đèn khác tắt).

4. Chu kỳ lặp lại.

|  |
| --- |
| // Khai báo các chân LED  const int redPin = 2; // Đèn đỏ  const int yellowPin = 3; // Đèn vàng  const int greenPin = 4; // Đèn xanh  void setup() {  // Khởi tạo các chân LED là OUTPUT  pinMode(redPin, OUTPUT);  pinMode(yellowPin, OUTPUT);  pinMode(greenPin, OUTPUT);  }  void loop() {  // Đèn đỏ sáng 3 giây, các đèn khác tắt  digitalWrite(redPin, HIGH); // Đèn đỏ sáng  digitalWrite(yellowPin, LOW); // Tắt đèn vàng  digitalWrite(greenPin, LOW); // Tắt đèn xanh  delay(3000); // Chờ 3 giây  // Đèn vàng sáng 1 giây, các đèn khác tắt  digitalWrite(redPin, LOW); // Tắt đèn đỏ  digitalWrite(yellowPin, HIGH); // Đèn vàng sáng  digitalWrite(greenPin, LOW); // Tắt đèn xanh  delay(1000); // Chờ 1 giây  // Đèn xanh sáng 7 giây, các đèn khác tắt  digitalWrite(redPin, LOW); // Tắt đèn đỏ  digitalWrite(yellowPin, LOW); // Tắt đèn vàng  digitalWrite(greenPin, HIGH); // Đèn xanh sáng  delay(7000); // Chờ 7 giây  } |

Câu 13. Cho thiết bị Arduino UNO, thực hiện kết nối 1 quang trở và 1 đèn LED đơn với thiết bị Arduino, trong đó một chân của quang trở được nối vào chân analog A1. Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau: Đọc giá trị của ánh sáng chiếu vào quang trở, nếu giá trị này vượt ngưỡng 550 thì đèn LED sẽ tắt, ngược lại dưới ngưỡng 550 đèn LED sẽ sáng. Đồng thời, hiển thị lên cửa sổ Serial Monitor thông báo “Giá trị ánh sáng:…. Đèn LED sáng hoặc tắt”

|  |
| --- |
| // Khai báo các chân  const int ldrPin = A1; // Chân analog A1 đọc giá trị từ quang trở  const int ledPin = 2; // Chân digital D2 điều khiển đèn LED  // Ngưỡng giá trị ánh sáng  const int threshold = 550; // Ngưỡng ánh sáng (550)  void setup() {  // Khởi tạo chân LED là OUTPUT  pinMode(ledPin, OUTPUT);    // Khởi tạo Serial Monitor  Serial.begin(9600);  }  void loop() {  // Đọc giá trị ánh sáng từ quang trở  int lightValue = analogRead(ldrPin);    // Kiểm tra giá trị ánh sáng và điều khiển đèn LED  if (lightValue > threshold) {  digitalWrite(ledPin, LOW); // Tắt LED nếu ánh sáng vượt ngưỡng  Serial.print("Giá trị ánh sáng: ");  Serial.print(lightValue);  Serial.println(" - Đèn LED tắt");  } else {  digitalWrite(ledPin, HIGH); // Sáng LED nếu ánh sáng dưới ngưỡng  Serial.print("Giá trị ánh sáng: ");  Serial.print(lightValue);  Serial.println(" - Đèn LED sáng");  }  // Chờ một chút trước khi đọc lại giá trị ánh sáng  delay(500);  } |

**Câu 14.** Cho thiết bị Arduino UNO, thực hiện kết nối 1 quang trở và 1 đèn RGB với Arduino, trong đó một chân của quang trở được nối vào chân analog A1. Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau:

Đọc giá trị của ánh sáng chiếu vào quang trở:

- Nếu giá trị ánh sáng <= 250: đèn RGB sáng màu đỏ

- Nếu 250 < giá trị ánh sáng <= 450: đèn RGB sáng màu xanh lá

- Nếu 450 < giá trị ánh sáng <= 550: đèn RGB sáng màu xanh dương

- giá trị chân analog>550: đèn sáng cả 3 màu

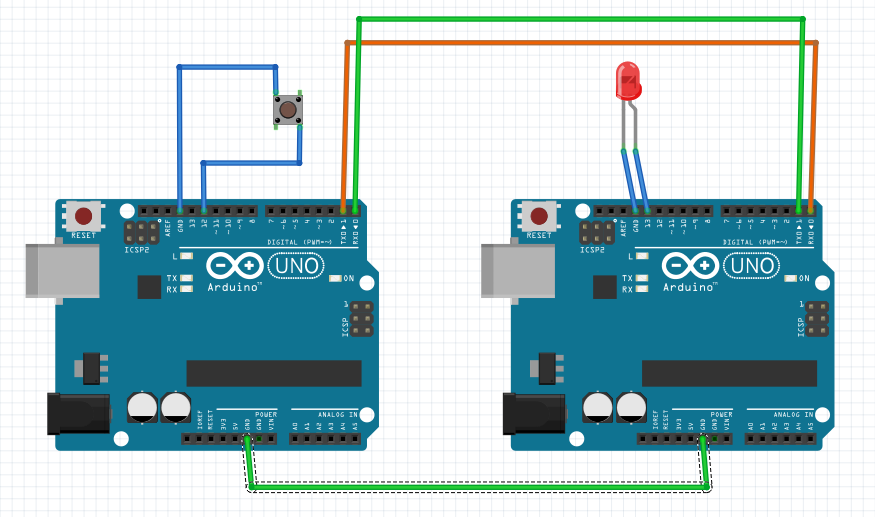
Đồng thời, hiển thị lên cửa sổ Serial Monitor thông báo “ Giá trị ánh sáng:…. Đèn RGB sáng màu:….”

|  |
| --- |
| // Khai báo các chân LED RGB  const int redPin = 9; // Chân Red của đèn RGB  const int greenPin = 10; // Chân Green của đèn RGB  const int bluePin = 11; // Chân Blue của đèn RGB  // Chân quang trở  const int ldrPin = A1; // Chân A1 đọc giá trị ánh sáng từ quang trở  void setup() {  // Khởi tạo các chân LED RGB là OUTPUT  pinMode(redPin, OUTPUT);  pinMode(greenPin, OUTPUT);  pinMode(bluePin, OUTPUT);    // Khởi tạo Serial Monitor  Serial.begin(9600);  }  void loop() {  // Đọc giá trị ánh sáng từ quang trở  int lightValue = analogRead(ldrPin);  // In giá trị ánh sáng ra cửa sổ Serial Monitor  Serial.print("Giá trị ánh sáng: ");  Serial.print(lightValue);  // Kiểm tra và điều khiển màu sắc của đèn RGB  if (lightValue <= 250) {  // Nếu ánh sáng <= 250, đèn RGB sáng màu đỏ  analogWrite(redPin, 255); // Sáng màu đỏ  analogWrite(greenPin, 0); // Tắt màu xanh lá  analogWrite(bluePin, 0); // Tắt màu xanh dương  Serial.println(" - Đèn RGB sáng màu đỏ");  }  else if (lightValue > 250 && lightValue <= 450) {  // Nếu 250 < ánh sáng <= 450, đèn RGB sáng màu xanh lá  analogWrite(redPin, 0); // Tắt màu đỏ  analogWrite(greenPin, 255); // Sáng màu xanh lá  analogWrite(bluePin, 0); // Tắt màu xanh dương  Serial.println(" - Đèn RGB sáng màu xanh lá");  }  else if (lightValue > 450 && lightValue <= 550) {  // Nếu 450 < ánh sáng <= 550, đèn RGB sáng màu xanh dương  analogWrite(redPin, 0); // Tắt màu đỏ  analogWrite(greenPin, 0); // Tắt màu xanh lá  analogWrite(bluePin, 255); // Sáng màu xanh dương  Serial.println(" - Đèn RGB sáng màu xanh dương");  }  else {  // Nếu ánh sáng > 550, đèn RGB sáng cả 3 màu  analogWrite(redPin, 255); // Sáng màu đỏ  analogWrite(greenPin, 255); // Sáng màu xanh lá  analogWrite(bluePin, 255); // Sáng màu xanh dương  Serial.println(" - Đèn RGB sáng cả 3 màu");  }  // Chờ một chút trước khi đọc lại giá trị ánh sáng  delay(500);  } |

**Câu 15.** Cho thiết bị Arduino UNO, thực hiện kết nối 1 quang trở và 1 còi (buzzer) với thiết bị Arduino, trong đó chân số 2 của quang trở được nối vào chân analog A0. Hãy thiết kế mạch và viết chương trình lập trình điều khiển cho thiết bị Arduino theo yêu cầu sau: Khi ánh sáng chiếu vào quang trở nhỏ hơn 500 sẽ kích hoạt hệ thống còi buzzer kêu, ngược lại còi không kêu . Đồng thời, hiển thị giá trị của ánh sáng chiếu vào quang trở và báo còi đang kêu hay đang tắt lên cửa sổ Serial Monitor.

|  |
| --- |
| const int ldrPin = A0; // Chân analog kết nối quang trở  const int buzzerPin = 8; // Chân digital điều khiển buzzer  void setup() {  pinMode(buzzerPin, OUTPUT); // Thiết lập chân buzzer là output  Serial.begin(9600); // Khởi động Serial Monitor  }  void loop() {  int lightValue = analogRead(ldrPin); // Đọc giá trị ánh sáng  Serial.print("Giá trị ánh sáng: ");  Serial.print(lightValue);  if (lightValue < 500) {  digitalWrite(buzzerPin, HIGH); // Bật còi  Serial.println(" - Còi đang kêu");  } else {  digitalWrite(buzzerPin, LOW); // Tắt còi  Serial.println(" - Còi đang tắt");  }  delay(500); // Chờ 0.5 giây trước lần đọc tiếp theo  } |

**Câu 16.** Cho thiết bị Arduino UNO được đặt tên là thiết bị A và thiết bị B. Sinh viên tự nối dây để thiết bị A có khả năng gửi tín hiệu cho thiết bị B thông qua Serial. Thiết bị A có một nút nối vào chân 12, thiết bị B có một đèn LED nối vào chân 13. Lập trình sao cho khi bấm và giữ nút ở thiết bị A thì đèn ở thiết bị B sáng, nếu không giữ nút ở thiết bị A thì đèn ở thiết bị B không sáng



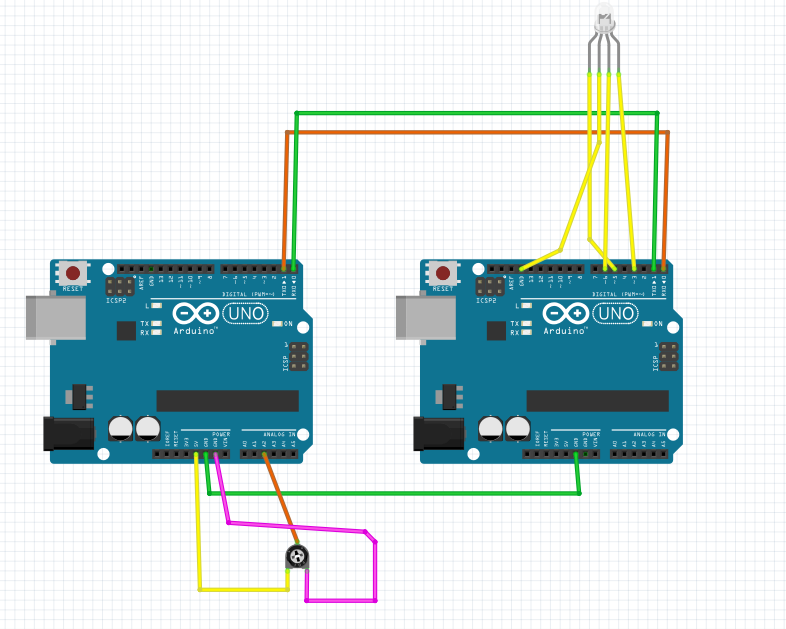
***Adruino A nhận tín hiệu (khi bấm nút)***

|  |
| --- |
| const int nut = 12;  void setup() {  pinMode(nut, INPUT\_PULLUP);  Serial.begin(9600);  }  void loop() {  if (digitalRead(nut) == LOW) {  Serial.println("1"); // Gửi tín hiệu bật  } else {  Serial.println("0"); // Gửi tín hiệu tắt  }  delay(100);  } |

***Arduino B - Nhận tín hiệu và điều khiển LED:***

|  |
| --- |
| const int led = 13;  void setup() {  pinMode(led, OUTPUT);  Serial.begin(9600);  }  void loop() {  if (Serial.available()) {  char data = Serial.read();  if (data == '1') {  digitalWrite(led, HIGH);  } else {  digitalWrite(led, LOW);  }  }  } |

**Câu 17.** Cho thiết bị Arduino UNO được đặt tên là thiết bị A và thiết bị B. Sinh viên tự nối dây để thiết bị A có khả năng gửi tín hiệu cho thiết bị B thông qua Serial. Thiết bị A có một biến trở có chân điều khiển nối vào chân A3, sinh viên tự nối các chân còn lại để biến trở hoạt động bình thường. Thiết bị B có một đèn LED RGB cho chân R, G và B nối lần lượt vào chân 5, 1, 3. Lập trình sao cho đèn LED RGB ở thiết bị B sáng màu xanh lá cây nếu a nhỏ hơn 513 và đèn sáng màu xanh da trời nếu a lớn hơn hoặc bằng 513 với a là giá trị thu được từ biến trở trên thiết bị A.



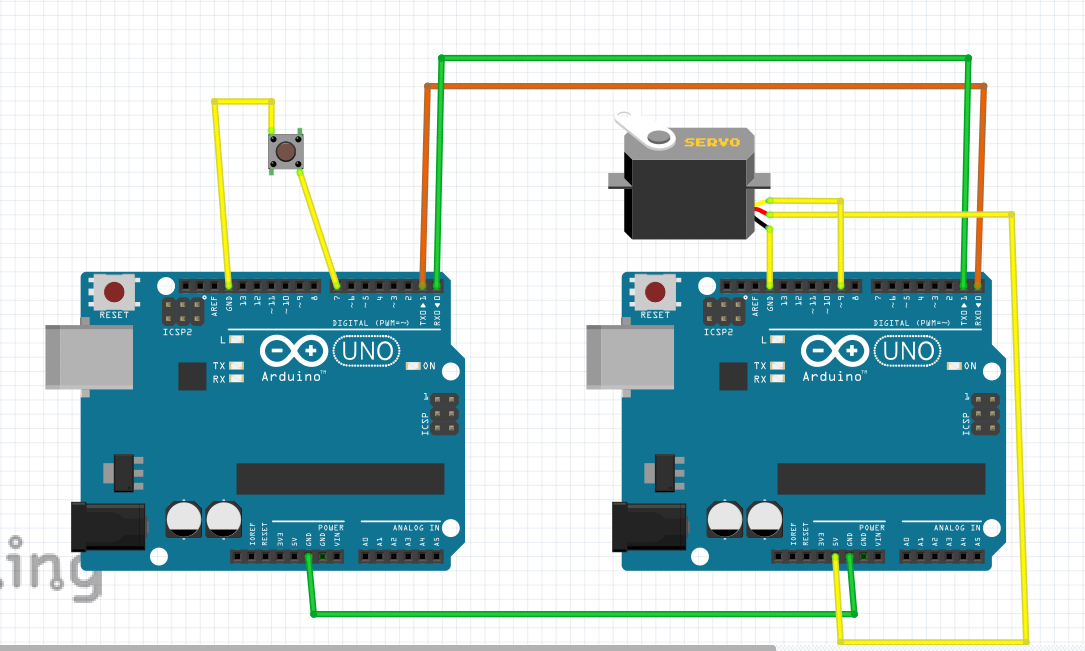
***Arduino A – Gửi giá trị biến trở***

|  |
| --- |
| const int bienTro = A3;  void setup() {  Serial.begin(9600);  }  void loop() {  int giaTri = analogRead(bienTro); // 0 - 1023  Serial.println(giaTri); // Gửi giá trị sang Arduino B  delay(100); // Gửi mỗi 100ms  } |

***Arduino B – Nhận và điều khiển LED RGB***

|  |
| --- |
| const int ledR = 5;  const int ledG = 6; // Dùng pin 6 thay vì 1 vì pin 1 thường dùng cho Serial  const int ledB = 3;  void setup() {  pinMode(ledR, OUTPUT);  pinMode(ledG, OUTPUT);  pinMode(ledB, OUTPUT);  Serial.begin(9600);  }  void loop() {  if (Serial.available()) {  String data = Serial.readStringUntil('\n');  int a = data.toInt();  if (a < 513) {  digitalWrite(ledR, LOW);  digitalWrite(ledG, HIGH); // Xanh lá  digitalWrite(ledB, LOW);  } else {  digitalWrite(ledR, LOW);  digitalWrite(ledG, LOW);  digitalWrite(ledB, HIGH); // Xanh dương  }  // In ra để debug  Serial.print("Giá trị biến trở: ");  Serial.print(a);  Serial.print(" | Đèn RGB: ");  if (a < 513)  Serial.println("Xanh lá");  else  Serial.println("Xanh dương");  }  } |

Câu 18. Cho thiết bị Arduino UNO được đặt tên là thiết bị A và thiết bị B. Sinh viên tự nối dây để thiết bị A có khả năng gửi tín hiệu cho thiết bị B thông qua Serial. Thiết bị A có nút bấm nối vào chân 7. Thiết bị B có động cơ Servo có chân điều khiển nối vào chân 9. Sinh viên tự nối các chân còn lại để đảm bảo động cơ hoạt động. Lập trình sao cho khi bấm và giữ nút ở thiết bị A thì động cơ Servo ở thiết bị B quay với giá trị góc 60o, khi không bấm nút thì động cơ Servo ở thiết bị B không kêu quay với giá trị mặc định là góc 0 độ.



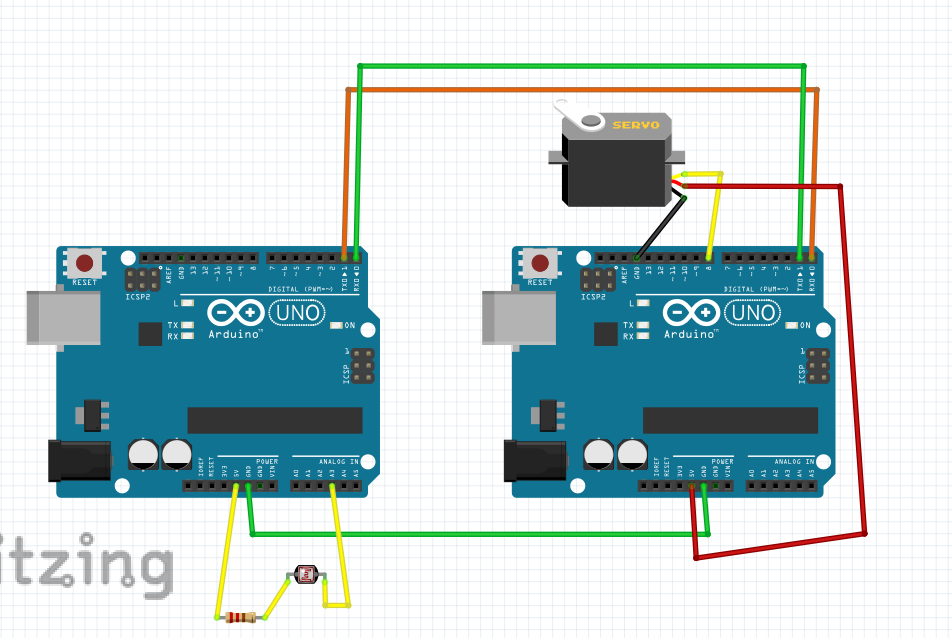
Code Aruino A (nút bấm)

|  |
| --- |
| const int buttonPin = 7;  void setup() {  pinMode(buttonPin, INPUT\_PULLUP); // nút thường nối xuống GND  Serial.begin(9600);  }  void loop() {  bool isPressed = digitalRead(buttonPin) == LOW;    if (isPressed) {  Serial.println("1");  } else {  Serial.println("0");  }  delay(100); // gửi dữ liệu mỗi 100ms  } |

Code Adruino B (Servo)

|  |
| --- |
| const int buttonPin = 7;  void setup() {  pinMode(buttonPin, INPUT\_PULLUP); // nút thường nối xuống GND  Serial.begin(9600);  }  void loop() {  bool isPressed = digitalRead(buttonPin) == LOW;    if (isPressed) {  Serial.println("1");  } else {  Serial.println("0");  }  delay(100); // gửi dữ liệu mỗi 100ms  } |

**Câu 19.** Cho thiết bị Arduino UNO được đặt tên là thiết bị A và thiết bị B. Sinh viên tự nối dây để thiết bị A có khả năng gửi tín hiệu cho thiết bị B thông qua Serial. Thiết bị A có quang trở nối vào chân A6. Thiết bị B có động cơ Servo có chân điều khiển nối vào chân 8. Sinh viên tự nối các chân còn lại để đảm bảo động cơ hoạt động. Lập trình sao cho động cơ Servo ở thiết bị B quay với giá trị góc a (đơn vị là độ), với a là giá trị đầu ra của hàm constrain có các tham số đầu vào đầu vào là giá trị thu được từ quang trở và giá trị chặn trên, chặn dưới lần lượt là 0 và 180.



Thiết bị A: Đọc quang trở và gửi dữ liệu

|  |
| --- |
| // THIẾT BỊ A  const int LDR\_PIN = A6;  void setup() {  Serial.begin(9600);  }  void loop() {  int lightValue = analogRead(LDR\_PIN); // Đọc giá trị ánh sáng  int angle = map(lightValue, 0, 1023, 0, 180); // Map về góc  angle = constrain(angle, 0, 180); // Giới hạn 0~180  Serial.println(angle); // Gửi sang thiết bị B  delay(100); // Gửi chậm để ổn định  } |

Thiết bị B: Nhận và điều khiển Servo

|  |
| --- |
| // THIẾT BỊ B  #include <Servo.h>  Servo myServo;  void setup() {  Serial.begin(9600);  myServo.attach(8); // Servo nối chân 8  }  void loop() {  if (Serial.available()) {  int angle = Serial.parseInt(); // Nhận dữ liệu  myServo.write(angle); // Quay servo theo góc  }  } |

**Câu 20.** Cho thiết bị Arduino UNO được đặt tên là thiết bị A và thiết bị B. Sinh viên tự nối dây để thiết bị A có khả năng gửi tín hiệu cho thiết bị B thông qua Serial. Thiết bị A có nút bấm nối vào chân 5. Thiết bị B có đèn LED có chân điều khiển nối vào chân 11. Lập trình sao cho khi bấm và giữ nút ở thiết bị A thì đèn LED ở thiết bị B sáng và tắt với tần số 2000 ms, khi không bấm nút thì đèn LED ở thiết bị B sáng và tắt với tần số 1000 ms.

**Sơ đồ : Tự mắc**

Thiết bị A: Gửi trạng thái nút bấm

|  |
| --- |
| // THIẾT BỊ A  const int buttonPin = 5;  void setup() {  pinMode(buttonPin, INPUT);  Serial.begin(9600);  }  void loop() {  int buttonState = digitalRead(buttonPin);  Serial.println(buttonState); // 1 nếu nhấn, 0 nếu không  delay(100); // Giảm tần số truyền  } |

Thiết bị B: Nhận tín hiệu và điều khiển LED nhấp nháy

|  |
| --- |
| // THIẾT BỊ B  const int ledPin = 11;  bool ledState = false;  unsigned long previousMillis = 0;  int blinkInterval = 1000; // Mặc định tắt/mở mỗi 1000ms  void setup() {  pinMode(ledPin, OUTPUT);  Serial.begin(9600);  }  void loop() {  if (Serial.available()) {  int buttonState = Serial.parseInt();  if (buttonState == 1) {  blinkInterval = 2000;  } else {  blinkInterval = 1000;  }  }  unsigned long currentMillis = millis();  if (currentMillis - previousMillis >= blinkInterval) {  previousMillis = currentMillis;  ledState = !ledState;  digitalWrite(ledPin, ledState);  }  } |

**Câu 21.** Cho thiết bị Arduino UNO. Cho màn hình LCD 16 x 2 kết hợp biến trở. Thí sinh tự thiết kế kết nối các chân của màn hình với các chân của thiết bị cũng như kết nối giữa LCD và biến trở để màn hình LCD hoạt động được. Cho hai biến trở P1 và quang trở P2 có chân điều khiển nối lần lượt vào chân A0 và A1. Trong 1 giây, thiết bị sẽ lấy giá trị biến trở R1 đưa vào biến a và giá trị quang trở P2 đưa vào biến b. Màn hình LCD sẽ hiển thị số a tính từ góc trái phía trên màn hình và a số b tính từ góc trái phía dưới màn hình.

|  |
| --- |
| #include <Wire.h>  #include <LiquidCrystal\_I2C.h>  // Khởi tạo màn hình LCD với địa chỉ I2C 0x27, và kích thước màn hình 16x2  LiquidCrystal\_I2C lcd(0x27, 16, 2);  int P1Pin = A0; // Biến trở P1 (R1) nối vào chân A0  int P2Pin = A1; // Quang trở P2 (LDR) nối vào chân A1  int a = 0; // Biến lưu giá trị của biến trở P1  int b = 0; // Biến lưu giá trị của quang trở P2  void setup() {  // Khởi tạo màn hình LCD  lcd.begin();  lcd.backlight(); // Bật đèn nền LCD  lcd.clear(); // Làm sạch màn hình  // In thông báo ban đầu  lcd.setCursor(0, 0);  lcd.print("Bien tro P1: ");  lcd.setCursor(0, 1);  lcd.print("Quang tro P2: ");  }  void loop() {  // Đọc giá trị của biến trở P1 (R1) từ chân A0 và quang trở P2 từ chân A1  a = analogRead(P1Pin);  b = analogRead(P2Pin);  // Chuyển đổi giá trị đọc được (từ 0 đến 1023) thành giá trị trong khoảng 0 đến 255  a = map(a, 0, 1023, 0, 255); // Điều chỉnh giá trị từ biến trở P1  b = map(b, 0, 1023, 0, 255); // Điều chỉnh giá trị từ quang trở P2  // Hiển thị các giá trị trên LCD  lcd.setCursor(0, 0);  lcd.print("Bien tro P1: ");  lcd.print(a); // Hiển thị giá trị từ biến trở P1  lcd.setCursor(0, 1);  lcd.print("Quang tro P2: ");  lcd.print(b); // Hiển thị giá trị từ quang trở P2  delay(1000); // Đợi 1 giây rồi làm mới màn hình  } |

Câu 22. Cho thiết bị Arduino UNO. Cho một thiết bị LED 7 đoạn. Cho nút bấm nối vào chân 8. Thí sinh tự thiết kế kết nối các chân của LED 7 đoạn với các chân của thiết bị (trừ chân số 8 nối vào nút bấm) để đảm bảo LED 7 đoạn hoạt động được. Cho số a = 0, khi người dùng bấm nút thì a cập nhật giá trị mới theo công thức a = (a + 1) % 10. Đèn LED 7 đoạn sẽ hiển thị giá trị a.

|  |
| --- |
| // Các chân nối với LED 7 đoạn  int ledPins[] = {2, 3, 4, 5, 6, 7, 8}; // Chân A, B, C, D, E, F, G của LED 7 đoạn  int buttonPin = 9; // Chân nút bấm  int a = 0; // Số hiển thị trên LED 7 đoạn  bool buttonState = false; // Trạng thái nút bấm  bool lastButtonState = false; // Trạng thái nút bấm trước đó  // Mảng chứa các mẫu hiển thị của LED 7 đoạn cho các số 0-9  int numTable[] = {  B1111110, // 0  B0110000, // 1  B1101101, // 2  B1111001, // 3  B0110011, // 4  B1011011, // 5  B1011111, // 6  B1110000, // 7  B1111111, // 8  B1111011 // 9  };  void setup() {  // Cấu hình các chân LED 7 đoạn là đầu ra  for (int i = 0; i < 7; i++) {  pinMode(ledPins[i], OUTPUT);  }  // Cấu hình chân nút bấm là đầu vào  pinMode(buttonPin, INPUT);    // Bắt đầu giao tiếp Serial để theo dõi trạng thái  Serial.begin(9600);  }  void loop() {  // Đọc trạng thái của nút bấm  buttonState = digitalRead(buttonPin);  // Kiểm tra nếu nút bấm thay đổi trạng thái  if (buttonState == HIGH && lastButtonState == LOW) {  // Khi nút được nhấn, cập nhật giá trị a  a = (a + 1) % 10;  displayNumber(a); // Hiển thị số lên LED 7 đoạn  Serial.print("Số hiển thị: ");  Serial.println(a);  delay(200); // Tránh nhấn quá nhanh  }  // Cập nhật trạng thái nút bấm trước đó  lastButtonState = buttonState;  }  // Hàm hiển thị số trên LED 7 đoạn  void displayNumber(int num) {  // Lấy mã nhị phân từ bảng numTable cho số cần hiển thị  int segments = numTable[num];    // Cập nhật các chân LED 7 đoạn theo mã nhị phân  for (int i = 0; i < 7; i++) {  int bitValue = (segments >> i) & 1; // Kiểm tra bit thứ i  digitalWrite(ledPins[i], bitValue); // Điều khiển LED theo bit  }  } |

**Câu 23.** Cho thiết bị Arduino UNO tương tác với máy tính qua Serial Monitor với tốc độ baud 9600. Cho 3 nút bấm B1, B2 kết nối lần lượt tới các chân điều khiển 3, 4. Cho giá trị a và b mặc định là 0. Khi bấm nút B1 thì a sẽ được gán giá trị ngẫu nhiên từ 0 đến 100. Khi Khi bấm nút B2 thì b sẽ được gán giá trị ngẫu nhiên từ 0 đến 100. Khi người dùng nhập ký tự “T” hoặc ký tự “t” thì in ra trên Serial Monitor tổng của a và b. Khi người dùng nhập ký tự “H” hoặc ký tự “h” thì in ra trên Serial Monitor hiệu của a và b.

Sơ đồ: Tự mắc

|  |
| --- |
| // Khai báo chân của các nút  const int buttonPin1 = 3; // Nút B1  const int buttonPin2 = 4; // Nút B2  // Khai báo các biến a và b  int a = 0;  int b = 0;  void setup() {  // Khởi tạo các chân nút là INPUT  pinMode(buttonPin1, INPUT);  pinMode(buttonPin2, INPUT);  // Khởi tạo Serial Monitor với tốc độ 9600 bps  Serial.begin(9600);  }  void loop() {  // Kiểm tra nút B1 (Chân 3)  if (digitalRead(buttonPin1) == HIGH) {  a = random(0, 101); // Gán giá trị ngẫu nhiên cho a từ 0 đến 100  Serial.print("Nút B1 nhấn: a = ");  Serial.println(a);  delay(200); // Đảm bảo nút bấm chỉ tác động một lần  }  // Kiểm tra nút B2 (Chân 4)  if (digitalRead(buttonPin2) == HIGH) {  b = random(0, 101); // Gán giá trị ngẫu nhiên cho b từ 0 đến 100  Serial.print("Nút B2 nhấn: b = ");  Serial.println(b);  delay(200); // Đảm bảo nút bấm chỉ tác động một lần  }  // Kiểm tra nhập ký tự từ Serial Monitor  if (Serial.available() > 0) {  char input = Serial.read(); // Đọc ký tự nhập vào  if (input == 'T' || input == 't') {  int sum = a + b; // Tính tổng a và b  Serial.print("Tổng của a và b là: ");  Serial.println(sum);  }  else if (input == 'H' || input == 'h') {  int difference = a - b; // Tính hiệu a và b  Serial.print("Hiệu của a và b là: ");  Serial.println(difference);  }  }  } |

**Câu 24.** Cho thiết bị Arduino UNO. Cho màn hình LCD 16 x 2 kết hợp biến trở. Thí sinh tự thiết kế kết nối các chân của màn hình với các chân của thiết bị cũng như kết nối giữa LCD và biến trở để màn hình LCD hoạt động được. Cho quang trở có chân điều khiển nối lần lượt vào chân A4. Trong 1 giây, thiết bị sẽ lấy giá trị biến trở đưa vào biến a và sản sinh số ngẫu nhiên b. Màn hình LCD sẽ hiển thị số c là tổng của a và b tính từ góc trái phía trên màn hình và số d là hiệu của a và b tính từ góc trái phía dưới màn hình.

Sơ đồ: Tự mắc

|  |
| --- |
| #include <LiquidCrystal.h>  // Khởi tạo đối tượng LCD, với các chân nối từ Arduino đến LCD  LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // RS, EN, D4, D5, D6, D7  // Khai báo các chân  const int lightSensorPin = A4; // Chân A4 nối với quang trở  const int potPin = A0; // Chân A0 nối với biến trở  void setup() {  // Khởi tạo màn hình LCD  lcd.begin(16, 2); // Màn hình LCD 16x2  lcd.print("Start"); // Hiển thị thông báo ban đầu    // Khởi tạo Serial Monitor  Serial.begin(9600);  }  void loop() {  // Đọc giá trị từ biến trở  int a = analogRead(potPin); // Giá trị từ 0 đến 1023  // Đọc giá trị từ quang trở  int lightValue = analogRead(lightSensorPin); // Giá trị từ 0 đến 1023    // Chuyển đổi giá trị a và b để làm cho giá trị dễ hiểu  float aValue = map(a, 0, 1023, 0, 100); // Chuyển giá trị biến trở thành giá trị từ 0 đến 100  int bValue = random(0, 101); // Sinh số ngẫu nhiên từ 0 đến 100    // Tính tổng và hiệu  int c = aValue + bValue; // Tổng của a và b  int d = aValue - bValue; // Hiệu của a và b    // Hiển thị kết quả lên màn hình LCD  lcd.clear(); // Xóa màn hình trước khi hiển thị  lcd.setCursor(0, 0); // Đặt con trỏ ở dòng đầu tiên  lcd.print("Tong: ");  lcd.print(c); // Hiển thị tổng    lcd.setCursor(0, 1); // Đặt con trỏ ở dòng thứ hai  lcd.print("Hieu: ");  lcd.print(d); // Hiển thị hiệu  // In ra giá trị trên Serial Monitor (nếu cần kiểm tra thêm)  Serial.print("a = ");  Serial.print(aValue);  Serial.print(" b = ");  Serial.println(bValue);    // Đợi 1 giây trước khi đọc lại giá trị  delay |