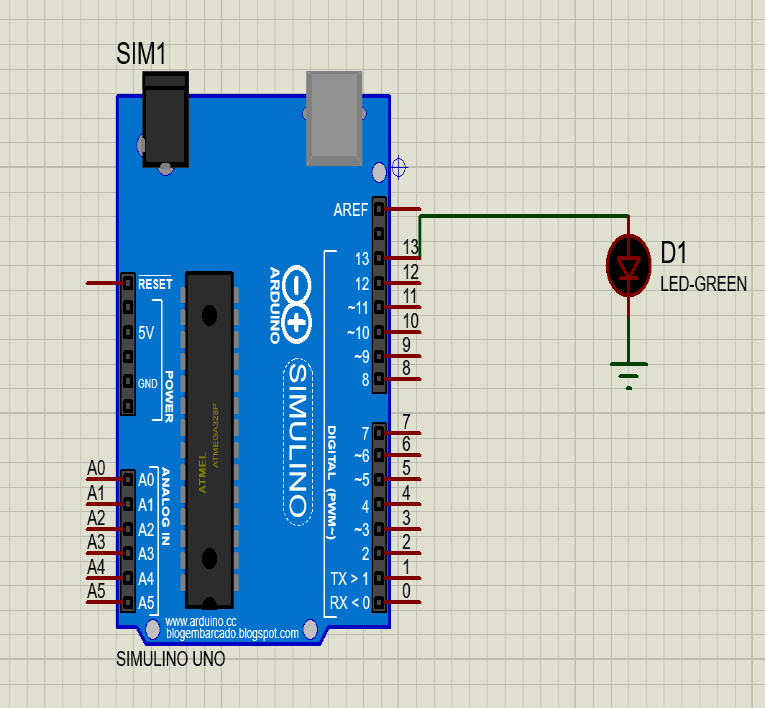
# BÀI 1: TẠO LED NHẤP NHÁY

## Mô tả

* + Thực hiện tạo 1 đèn LED nhấp nháy trên bo mạch Arduino trên phần mềm Protues.

## Sơ đồ mạch



Hình 1: Sơ đồ LED nhấp nháy trên Protues

## Cần chuẩn bị

* Một bo mạch Arduino Uno.
* Một đèn LED.

## Code để thực hiện

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | **Int** pin=13; | | | |  | void **setup**(){ | |  | **pinMode**(pin,OUTPUT); | |  | } | |  | void **loop**(){ | |  | **digitalWrite**(pin,HIGH); | |  | **delay**(200); | |  | **digitalWrite**(pin,LOW); | |  | **delay(**200); | |  | } | |

# BÀI 2: DÙNG NÚT ĐỂ BẬT LED

## Mô tả

* + Thực hiện thêm nút bấm và dùng nút để bật tắt LED.

## Sơ đồ mạch

Hình 2: Sơ đồ LED dùng nút bấm trên Protues

## Cần chuẩn bị

* 1 bo mạch Arduino Uno.
* 1 LED.
* 2 điện trở (Resistor).
* 1 nút bấm.

## Code để thực hiện

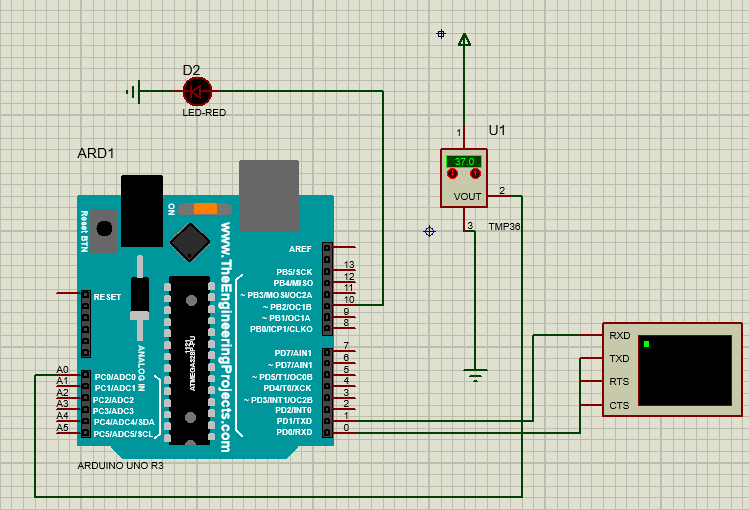
|  |
| --- |
| **int** x=0;  void **setup**(){  **pinMode**(2,INPUT);  **pinMode**(13, OUTPUT);}  void **loop**(){  x=**digitalRead**(2);  **if**(x==HIGH) {  **digitalWrite**(13,HIGH);}  **else** {  **digitalWrite**(13,LOW);  }  **delay**(1000)} |

# BÀI 3: LED SÁNG DỰA TRÊN NHIỆT ĐỘ

## Mô tả

* + Sử dụng cảm biến nhiệt độ để đo nhiệt độ và dựa trên nhiệt độ nhất định để làm sáng đèn LED.

## Sơ đồ mạch



Hình 3: Sơ đồ LED sáng dựa trên nhiệt độ

## Cần chuẩn bị

* 1 bo mạch Arduino Uno R3.
* 1 LED.
* 1 cảm biến nhiệt độ TMP36.
* 1 màn hình Terminal.

## Code để thực hiện

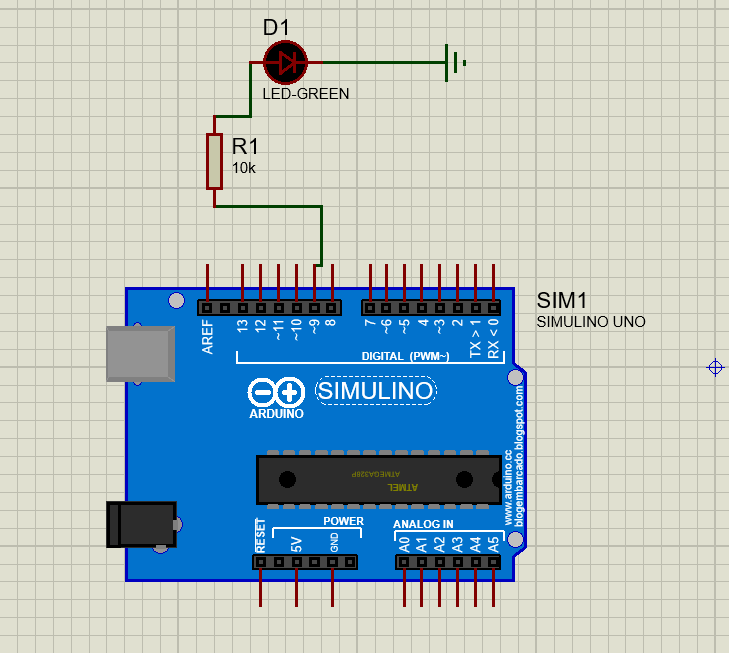
|  |
| --- |
| void **setup**(){  **pinMode**(10,OUTPUT);  **Serial.begin**(9600);}  void **loop**(){  **int** GiaTri = **analogRead**(A0);  **int** nhietdo=**map**(GiaTri,20,358,-40,125);  **Serial.print**(nhietdo);  **if** (nhietdo>37) **digitalWrite**(10,HIGH);  **else digitalWrite**(10,LOW);  **delay**(1000);  } |

# BÀI 4: LED SÁNG DẦN

## Mô tả

* + Thực hiện làm 1 đèn LED sáng dần.

## Sơ đồ mạch



Hình 4 Sơ đồ LED sáng dần

## Cần chuẩn bị

* 1 bo mạch Arduino Uno.
* 1 điện trở (Resistor).
* 1 đèn LED.

## Code để thực hiện

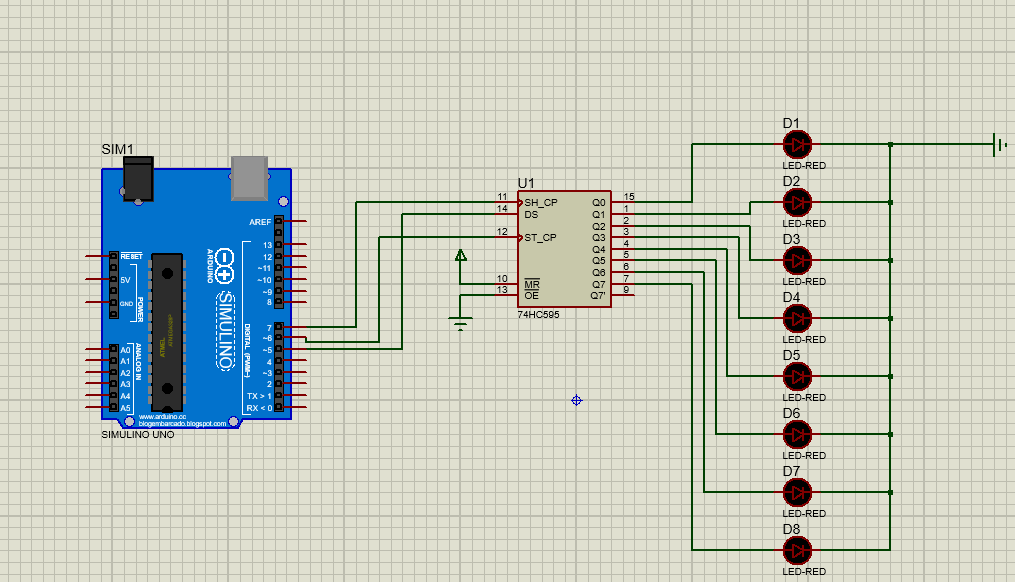
|  |
| --- |
| **int** brightness = 0;  void **setup**(){  **pinMode**(9,OUTPUT);}  void **loop**(){  **for** (brightness = 0; brightness <=255; brightness +=5){  **analogWrite**(9,brightness);  **delay**(30);}  **for**(brightness = 255; brightness >=0; brightness -=5){  **analogWrite**(9,brightness);  **delay**(30);}  } |

# BÀI 5: NHÁY 8 ĐÈN LED

## Mô tả

* + Thực hiện nháy 8 đèn LED 1 cách tuần tự

## Sơ đồ mạch



Hình 5: Sơ đồ nháy 8 LED trên mạch Arduino

## Cần chuẩn bị

* 1 bo mạch Arduino Uno
* 1 Thanh ghi dịch 74HC595 (Shift Register)
* 8 đèn LED

## Code chuẩn bị

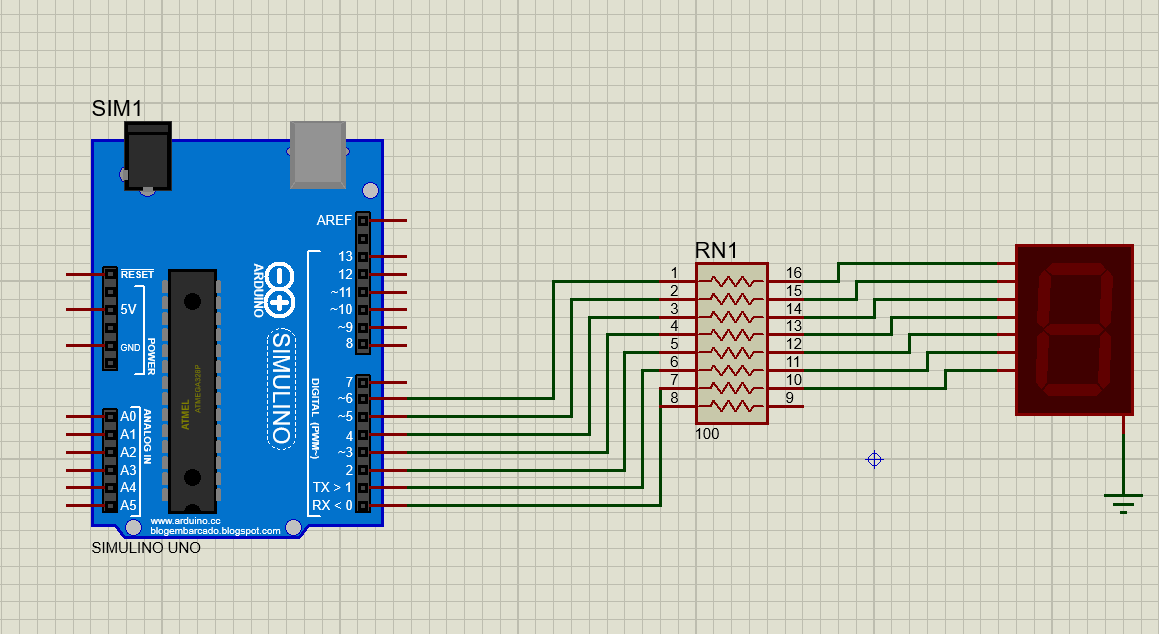
|  |
| --- |
| #define \_clock 7  #define \_latch 6  #define \_data 5  void **setup**() {  **pinMode**(\_latch,OUTPUT); **pinMode**(\_clock,OUTPUT); **pinMode**(\_data,OUTPUT);}  void **loop**() {  **for**(**int** i=0; i<256;i++){  **digitalWrite**(\_latch,LOW);  **shiftOut**(\_data,\_clock,**LSBFIRST**,i);  **digitalWrite**(\_latch,HIGH);  **delay**(500); } |

# BÀI 6: LED 7 ĐOẠN

## Mô tả

* + Thực hiện hiển thị số từ 0 đến 9 trên màn hình LED 7 đoạn

## Sơ đồ mạch



Hình 6:Sơ đồ LED 7 đoạn trên mạch Arduino Uno

## Cần chuẩn bị

* 1 bo mạch Arduino Uno
* 1 điện trở thanh 16 chân (8 Ways Resistor Network)
* 1 bảng LED 7 đoạn

## Code chuẩn bị

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

# BÀI 7: CẢM BIẾN SÓNG ÂM

## Mô tả

## Sử dụng cảm biến sóng âm HC-SR04 để tiến hành đo khoảng cách và cho kết quả chính xác trong khoảng cách từ 2 ->300m.

## Sơ đồ mạch

Hình 7:Sơ đồ cảm biến sóng âm trên bo mạch Arduino

## Cần chuẩn bị

* 1 bo mạch Arduino
* 1 cảm biến sóng âm HC-SR04
* 1 màn hình Serial để hiển thị khoảng cách

## Code chuẩn bị

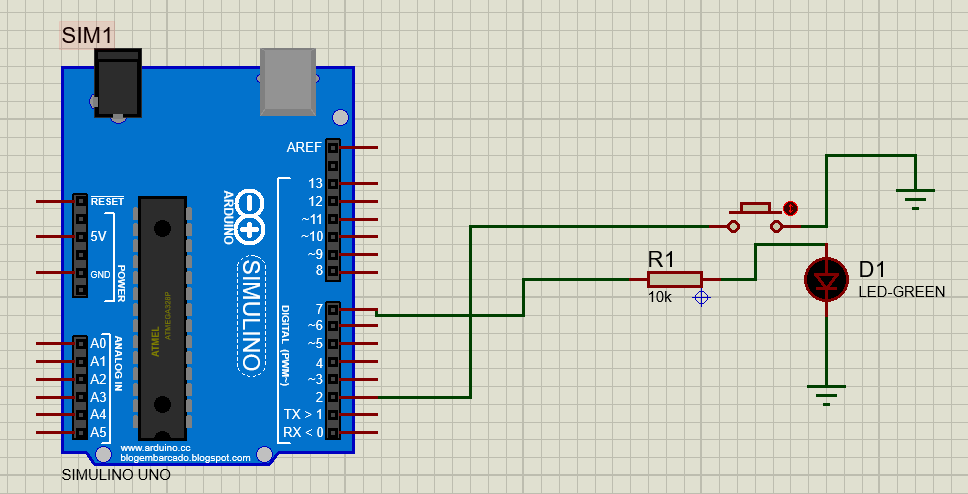
|  |
| --- |
| **const int** trig = 8; // chân trig của HC-SR04  **const int** echo = 7; // chân echo của HC-SR04  void **setup**(){  **Serial.begin**(9600); **pinMode**(trig,OUTPUT); **pinMode**(echo,INPUT); }  void **loop**(){  **unsigned long** duration;  **int** distance;  **digitalWrite**(trig,0);  **delayMicroseconds**(2);  **digitalWrite**(trig,1);  **delayMicroseconds**(5);  **digitalWrite**(trig,0);  // Đo độ rộng xung HIGH ở chân echo.  duration = **pulseIn**(echo,HIGH);  distance = **int**(duration/2/29.412);  /\* In kết quả ra Serial Monitor \*/  **Serial.print**(distance);  **Serial.println**("cm");  **delay**(200);} |

# BÀI 8: PHÍM NGẮT

## Mô tả

* + Cho 1 đèn LED đang sáng và khi ấn giữ nút làm đèn tắt cho đến khi thả nút ra

## Sơ đồ mạch



Hình 8:Sơ đồ phím ngắt trên bo mạch Arduino Uno

## Cần chuẩn bị

* 1 bo mạch Arduino Uno
* 1 điện trở
* 1 nút bấm
* 1 đèn LED

## Code chuẩn bị

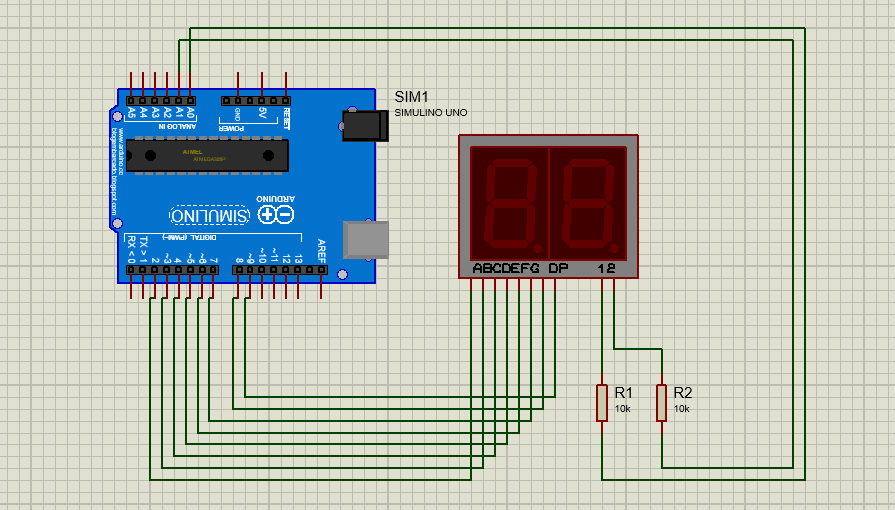
|  |
| --- |
| **const int** buttonPin = 2;  **const int** ledPin = 7;  **volatile int** buttonState = 0;  void **setup**() {  **pinMode**(ledPin, OUTPUT);  **pinMode**(buttonPin, INPUT);  **attachInterrupt**(**digitalPinToInterrupt**(buttonPin), **pin\_ISR**, **CHANGE**);}  void **loop**() {}  void **pin\_ISR**() {  buttonState = **digitalRead**(buttonPin);  **digitalWrite**(ledPin, buttonState);} |

# BÀI 9: HIỂN THỊ SỐ TỪ 0 ĐẾN 99 TRÊN MÀN HÌNH LED 7 ĐOẠN

## Mô tả

* + Hiển thị số bắt đầu từ 0 đến 99 trên màn hình LED 7 đoạn

## Sơ đồ mạch



Hình 9:Sơ đồ LED 7 đoạn trên bo mạch Arduino

## Cần chuẩn bị

* 1 bo mạch Arduino Uno
* 1 màn hình hiển thị 2 LED 7 đoạn
* 2 điện trở

## Code chuẩn bị

|  |
| --- |
| **byte** segValue[10][7] = {  {0,0,0,0,0,0,1}, //0  {1,0,0,1,1,1,1}, //1  {0,0,1,0,0,1,0}, //2  {0,0,0,0,1,1,0}, //3  {1,0,0,1,1,0,0}, //4  {0,1,0,0,1,0,0}, //5  {0,1,0,0,0,0,0}, //6  {0,0,0,1,1,1,1}, //7  {0,0,0,0,0,0,0}, //8  {0,0,0,0,1,0,0} //9 };  **byte** segPin[8]={2,3,4,5,6,7,8,9};  **byte** digitPin[2] = {A0,A1};  **unsigned long** readTime=0;  **int** d1 = 0;  **int** d2 = 0;  void **setup**() {  **for**(int i=0;i<10;i++){  **pinMode**(segPin[i], OUTPUT);}  **pinMode**(digitPin[0], OUTPUT);  **pinMode**(digitPin[1], OUTPUT);  **digitalWrite**(digitPin[0], LOW);  **digitalWrite**(digitPin[1], LOW);  }  void **loop**() {  readTime = **millis**()/1000;  d1 = **readTime**%10;  d2 = (**readTime**/10)%10;  segOutput(1,d1,1);  if(**readTime**>=10) segOutput(0,d2,1); }  void segClear(){  **for**(int i=0;i<8;i++){  **digitalWrite**(segPin[i], HIGH);}  }  void segOutput(int d, int Number, int dp){  segClear();  **digitalWrite**(digitPin[d], HIGH);  **for**(int i=0;i<7;i++){  **digitalWrite**(segPin[i], segValue[Number][i]);}  **digitalWrite**(segPin[7], dp);  **delayMicroseconds**(1000);  **digitalWrite**(digitPin[d], LOW);  } |

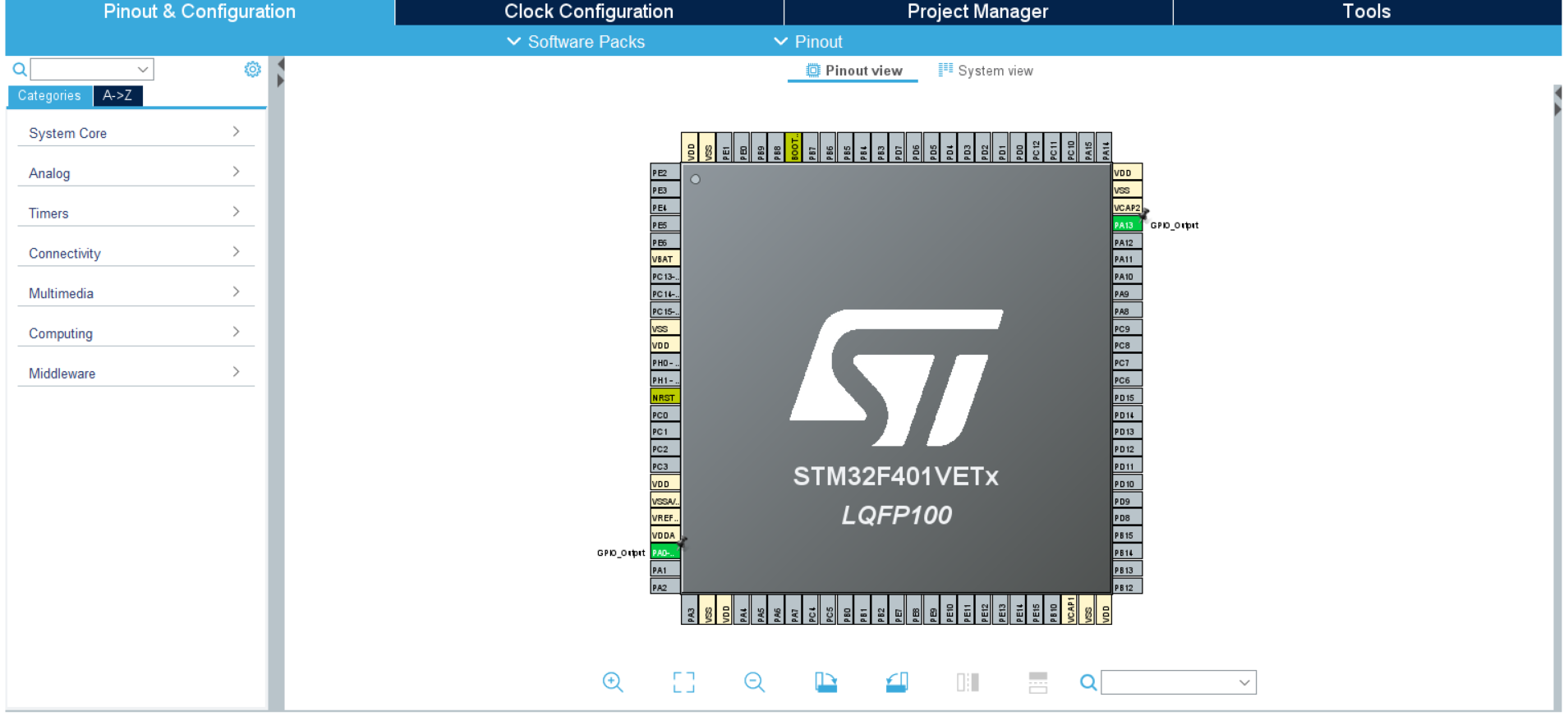
# LẬP TRÌNH TRÊN BOARD MẠCH STM32

# Bài 1: Tạo nút bật LED

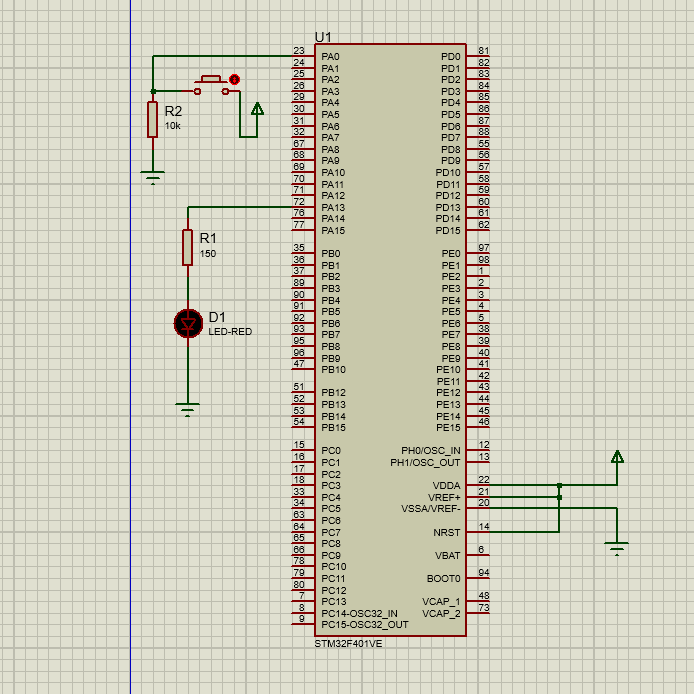
## Mô tả

## Dùng nút bấm để làm sáng đèn LED trên bo mạch STM32F401VE thông qua phần mềm STM32Cube và KeilC

## Sơ đồ STM32F401VE trong phần mềm STM32Cube



## Sơ đồ mạch qua Proteus



## Cần chuẩn bị

* STM32F401VE trong phần mềm STM32CubeMX để tạo code
* STM32F401VE trong proteus
* 2 điện trở
* 1 nút bấm
* 1 đèn LED

## Cấu hình mạch STM32F401VE

* Thiết lập chân PA0 là Output(GPIO\_Output)
* Thiết lập chân PA13 là Output(GPIO\_Output)

## Code chuẩn bị

|  |
| --- |
| while (1)  {  GPIO\_PinState pin0State =HAL\_GPIO\_ReadPin(GPIOA, GPIO\_PIN\_0);  if(pin0State == GPIO\_PIN\_SET)  {  //Bâtt Led  HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_13,GPIO\_PIN\_SET);  }  else{ //Tătt Led  HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_13,GPIO\_PIN\_RESET);  }  }  } |

# Bài 2: Nút Ngắt LED

## Mô tả

* Thiết lập ngắt, khi giữ nút thì đèn LED tắt và thả ra thì đen LED sẽ sáng

## Sơ đồ STM32F401VE trong phần mềm STM32Cube

## 

## Sơ đồ mạch qua Proteus

## 

## Cần chuẩn bị

* STM32F401VE trong phần mềm STM32CubeMX để tạo code
* STM32F401VE trong proteus
* 1 nút bấm
* 4 điện trở
* 3 đèn LED

## Code chuẩn bị

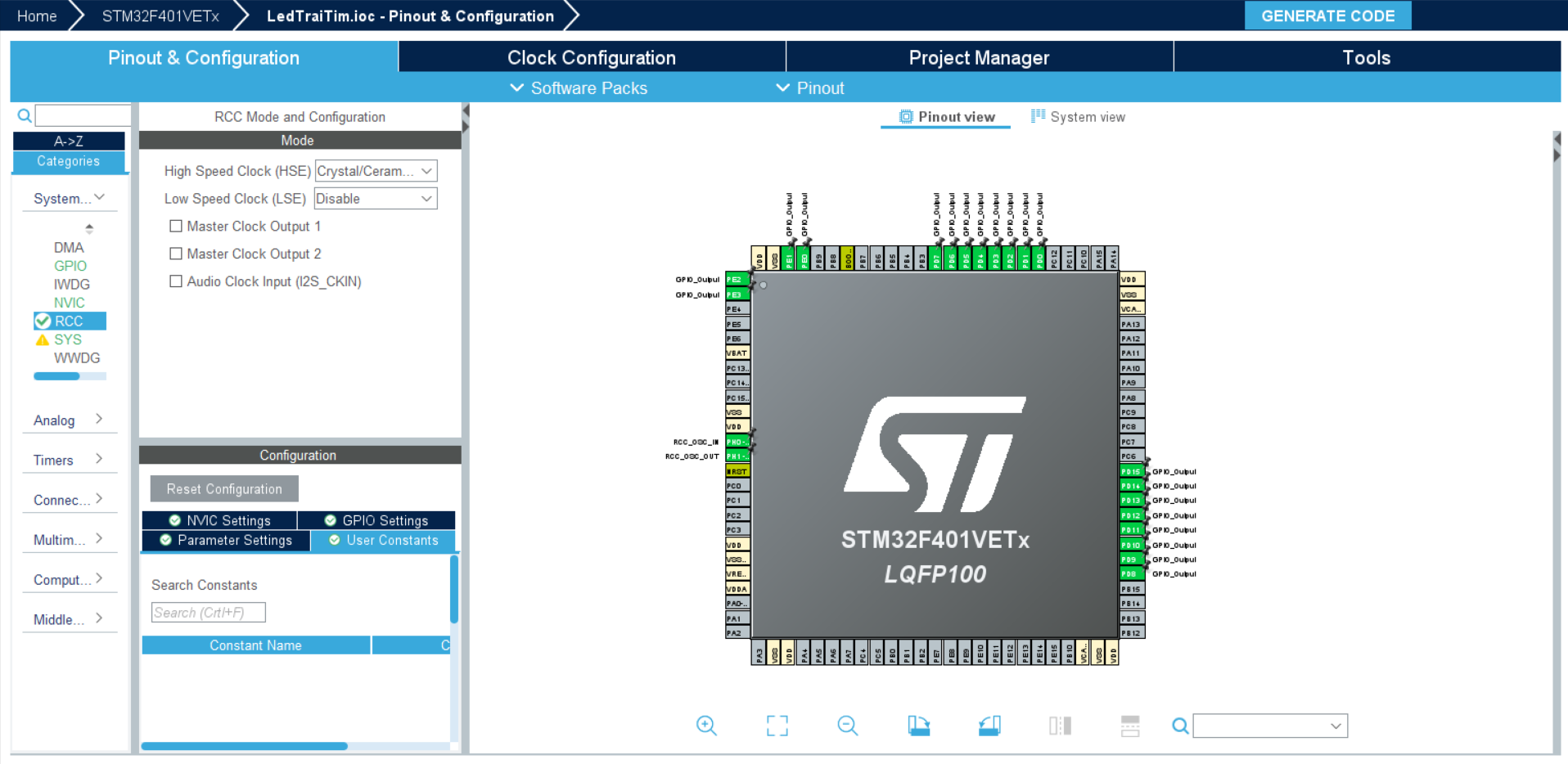
|  |
| --- |
| while (1)  {  HAL\_GPIO\_TogglePin(GPIOD,GPIO\_PIN\_0|GPIO\_PIN\_1|GPIO\_PIN\_2);  HAL\_Delay(1000);  }  }//Het ham main  //chuong trinh ngat  void HAL\_GPIO\_EXTI\_Callback(uint16\_t GPIO\_Pin)  {  if(GPIO\_Pin == GPIO\_PIN\_0)  {  HAL\_GPIO\_WritePin(GPIOA,GPIO\_PIN\_10,GPIO\_PIN\_SET);  }  } |

# BÀI 3:LED TRÁI TIM

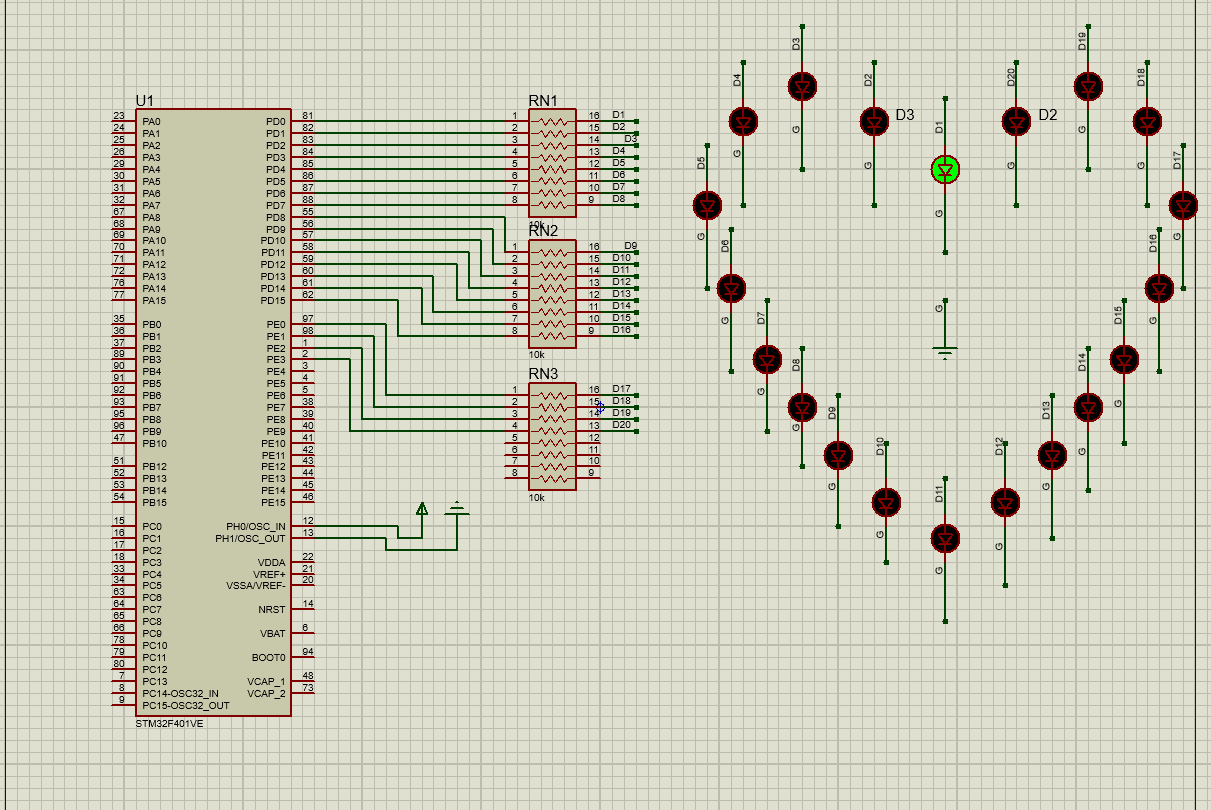
## Mô tả

* Sắp xếp các đèn LED thành hình trái tim và cho sáng lần lượt từ trái sang phải

## Sơ đồ STM32F401VE trong phần mềm STM32Cube



## Sơ đồ mạch trong Protues



## Cần Chuẩn bị

* STM32F401VE trong phần mềm STM32CubeMX để tạo code
* STM32F401VE trong proteus
* 3 điện trở thanh 16 chân (8 Ways Resistor Network)
* 20 Đèn LED

## Code Chuẩn bị

|  |
| --- |
| void Sangden();  void Tatden();  void Sanglanluot1();  void Sanglanluot2();  int main(void)  {  HAL\_Init();  SystemClock\_Config();  MX\_GPIO\_Init();  while (1)  {  Sangden();  HAL\_Delay(100);  Tatden();  HAL\_Delay(100);  Sanglanluot1();  HAL\_Delay(100);  Sanglanluot2();    }  }  void Sangden(){  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);  }  void Tatden(){  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);  }  void Sanglanluot1(){  Tatden();  HAL\_Delay(100);  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);  }  void Sanglanluot2(){  Tatden();  HAL\_Delay(100);    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);  } |