Khánh Hòa - 2021

BÁO CÁO LẬP TRÌNH NHÚNG

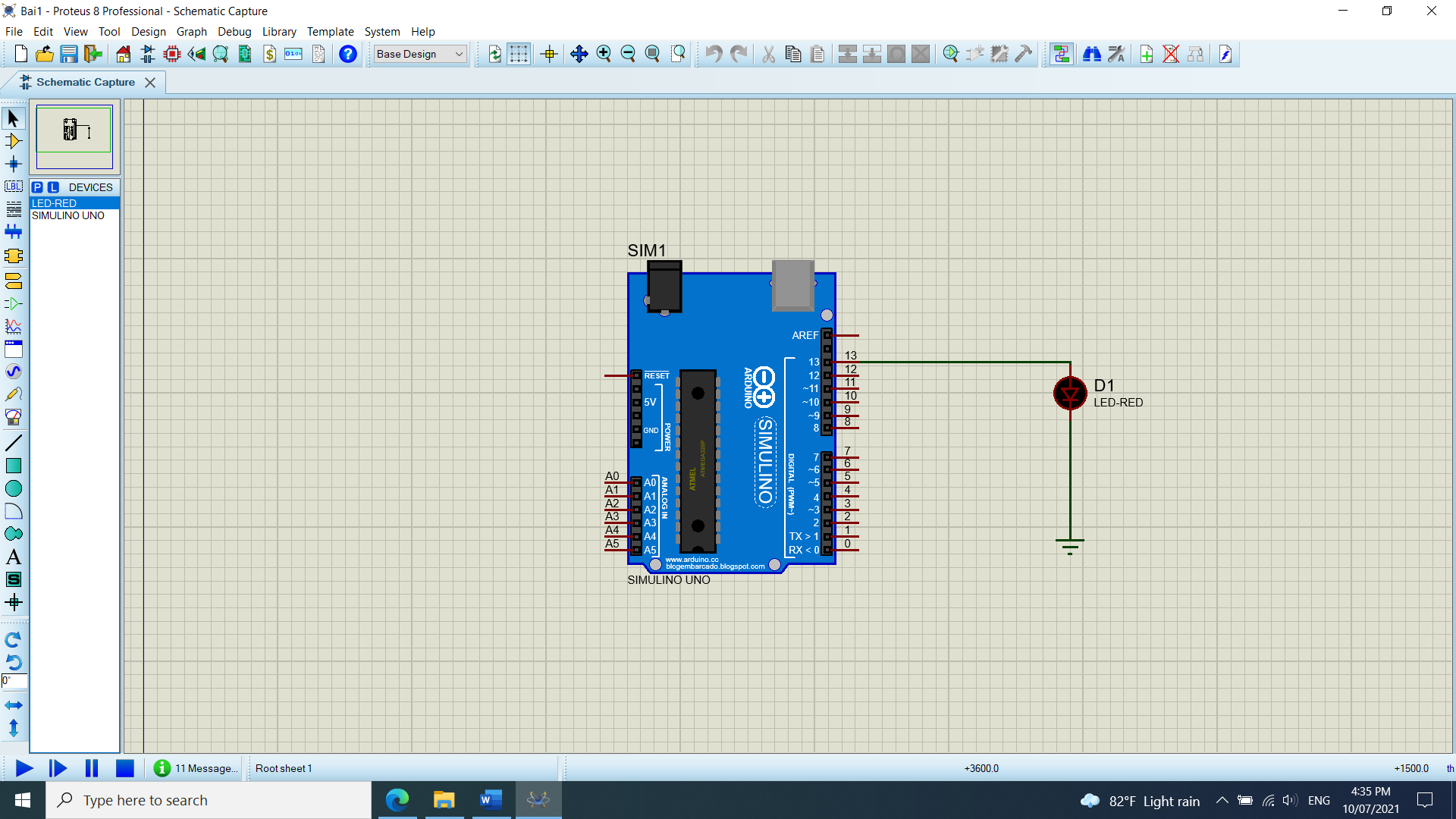
Sinh viên thực hiện: Nguyễn Thành Hưng Mssv:61131801

# Bài 1. Nháy đèn Led

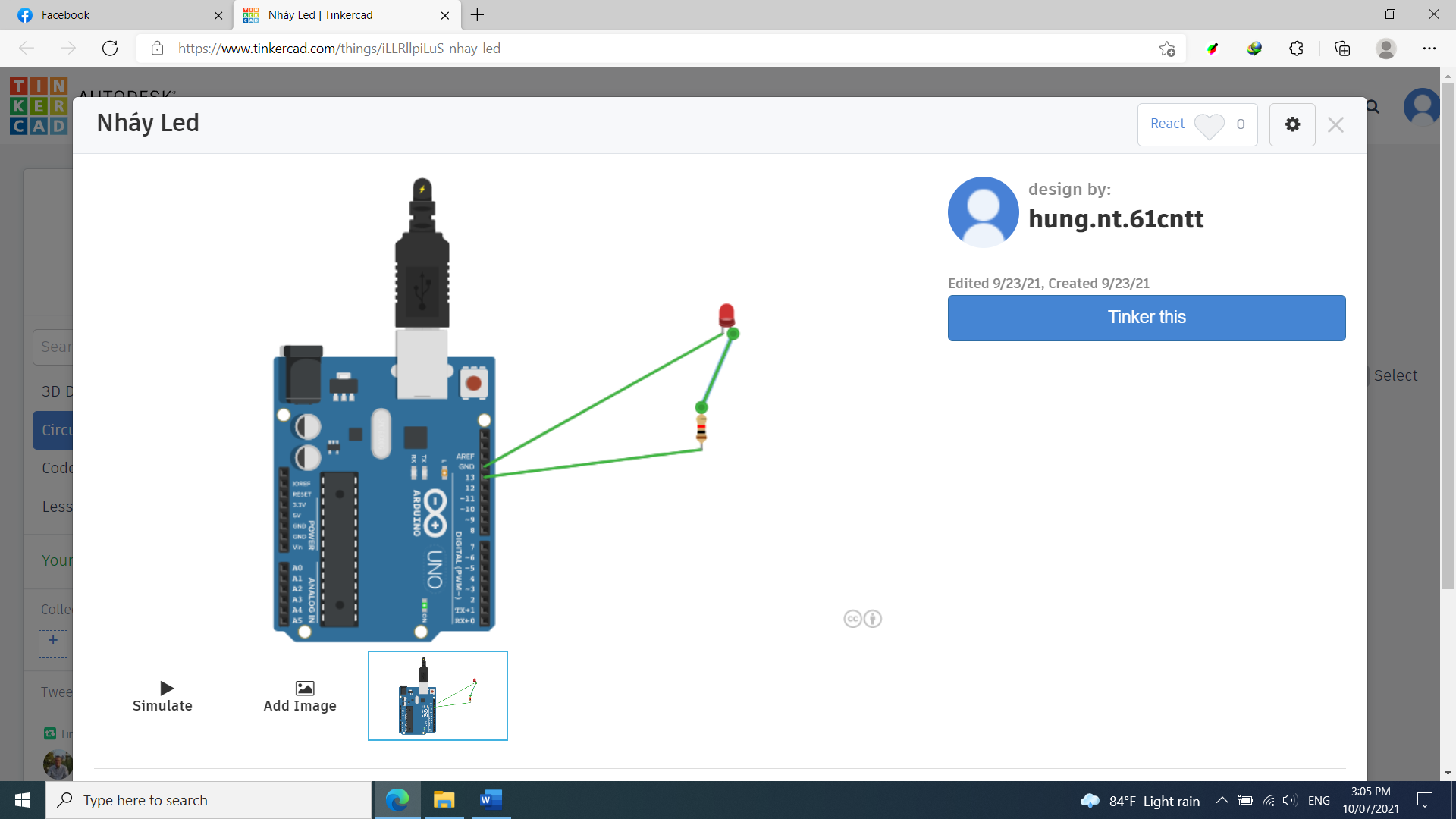
## Mô tả

Hệ thống được thiết kế thực hiện việc lập trình điều kiển nháy đèn Led trong thời gian 1 giây, đèn Led được kết nối vào cổng số 13 của board mạch Arduino.

## Sơ đồ thiết kế



Hình 1. Sơ đồ mạch Proteus



Hình 2. Sơ đồ mạch Tinkercad

## Đặc điểm của linh kiện

* 1 mạch Arduino Uno
* 1 đèn Led-RED
* 1 điện trở: 100 Ω

## Code chương trình

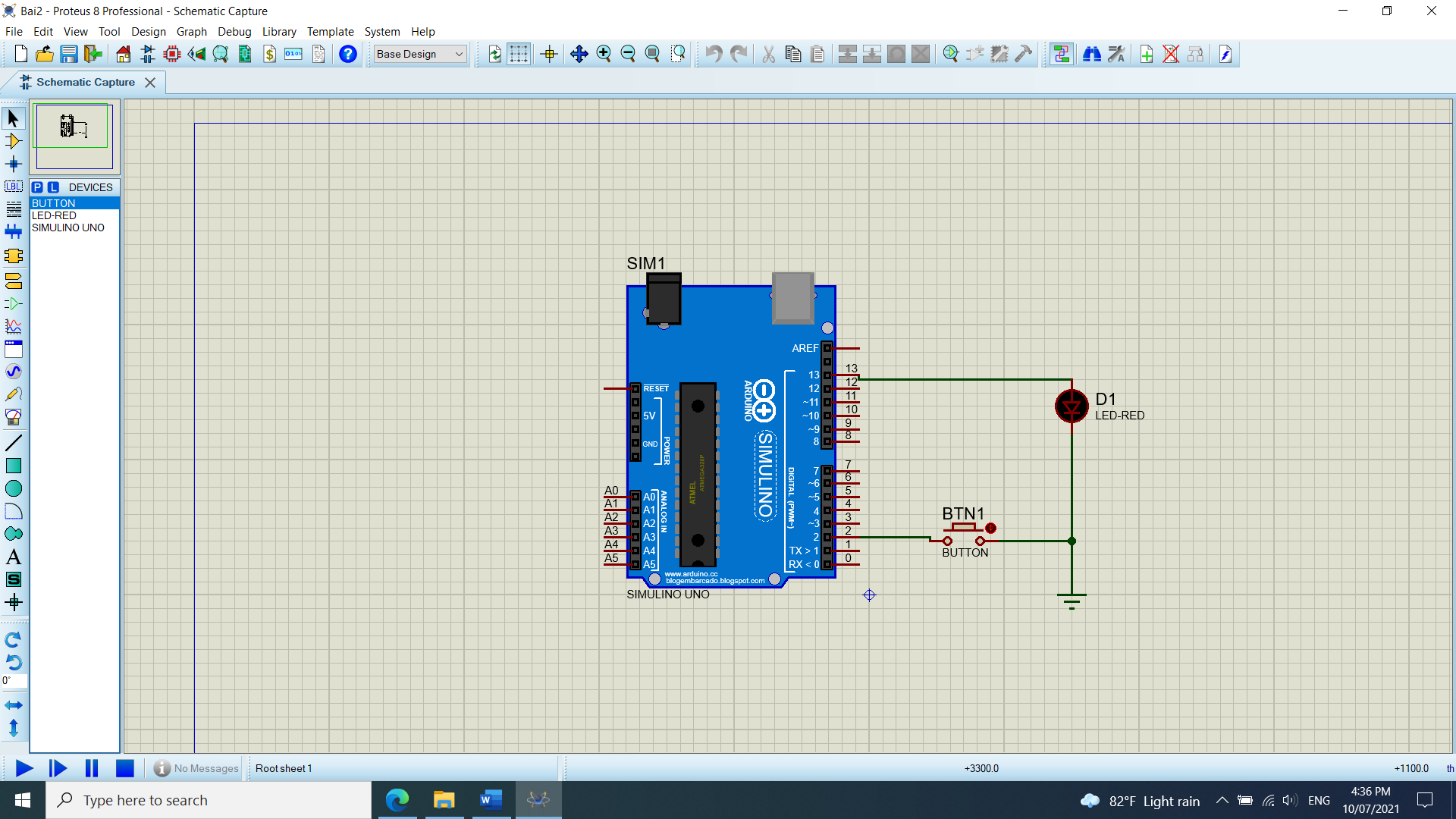
|  |
| --- |
| void **setup**() {  pinMode(13, OUTPUT);  }  void **loop**() {  digitalWrite(13, HIGH); // Bật Led  delay(1000); // Để Led sáng 1 giây  digitalWrite(13, LOW); // Tắt Led  delay(1000); // Trong 1 giây  } |

# Bài 2. Nút bấm đè bật đèn Led

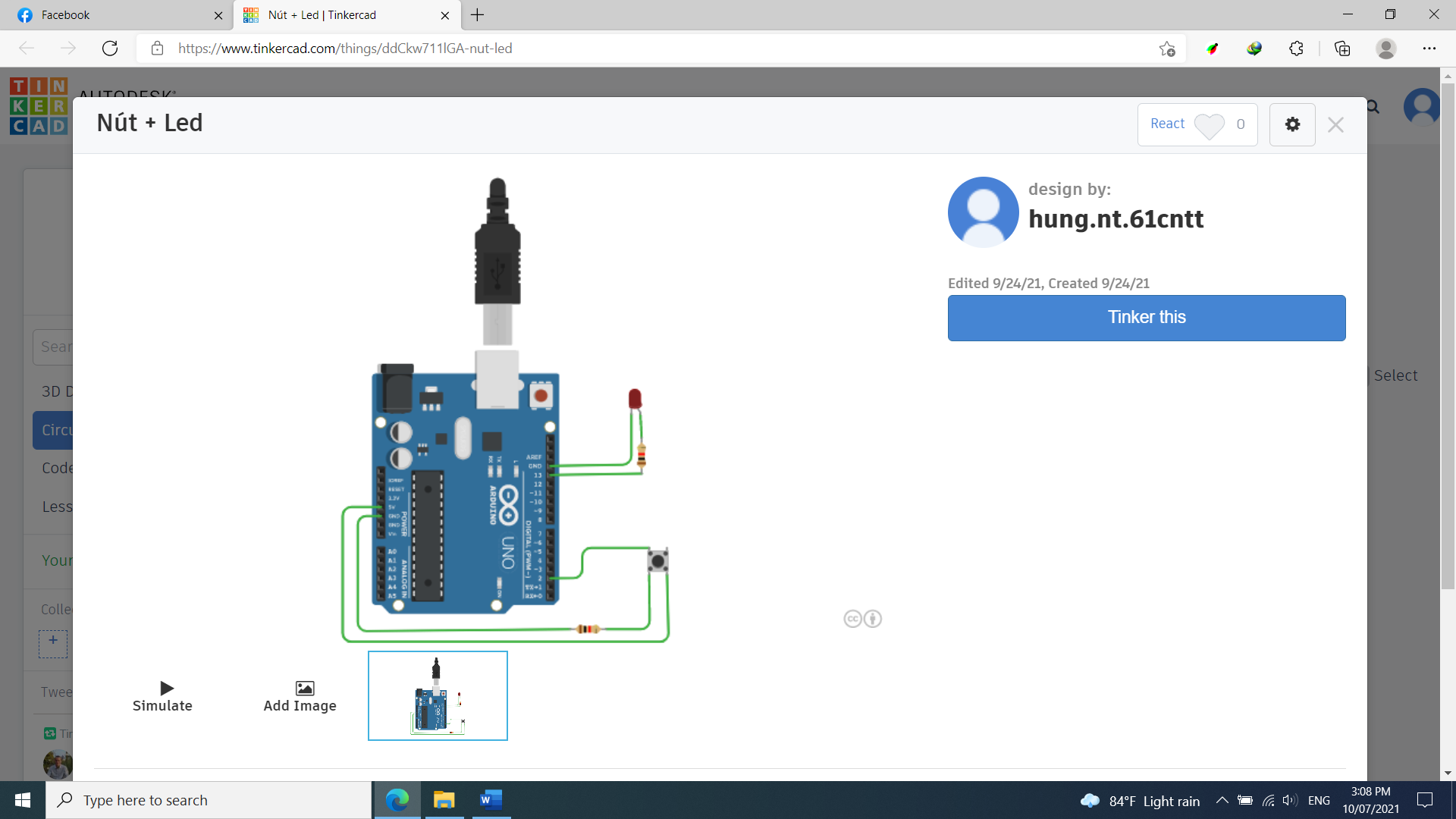
## Mô tả

Hệ thống được thiết kế thực hiện việc lập trình điều kiển bật/tắt đèn Led thông qua nút bấm đè, đèn Led được kết nối vào cổng số 13 của board mạch Arduino.

## Sơ đồ thiết kế



Hình 3. Sơ đồ mạch Proteus



Hình 4. Sơ đồ mạch Tinkercad

## Đặc điểm của linh kiện

* 1 đèn Led
* 1 mạch Arduino Uno
* 2 điện trở: 100 Ω
* 1 nút bấm

## Code chương trình

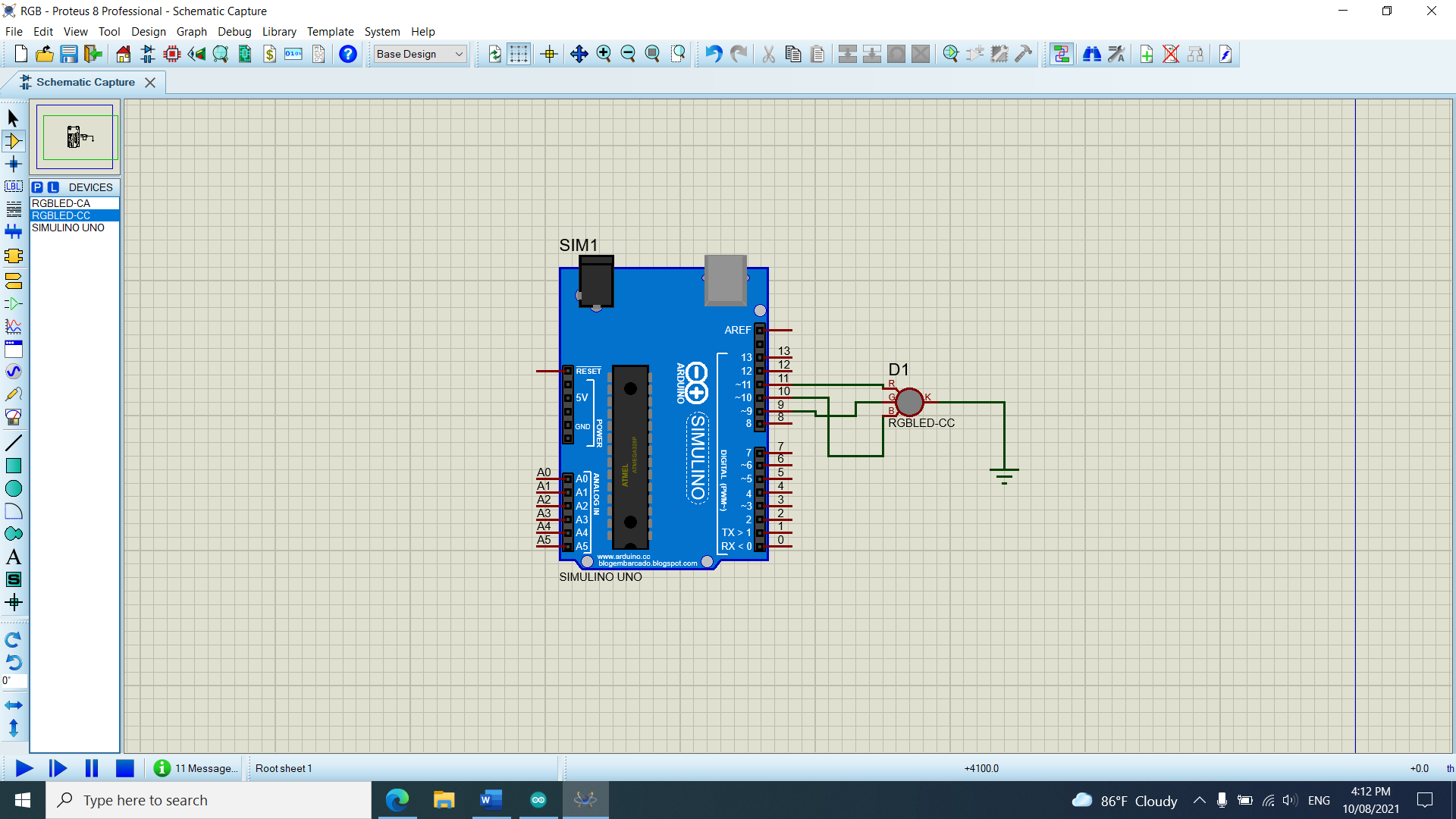
|  |
| --- |
| #define led 13  #define btn 2  void setup()  {  pinMode(btn,INPUT);  pinMode(led, OUTPUT);  }  void loop()  {  if(digitalRead(btn)==0)  {  digitalWrite(13, LOW);  }  else  {  digitalWrite(13, HIGH);  }  } |

# Bài 3. Nháy đèn LED RGB

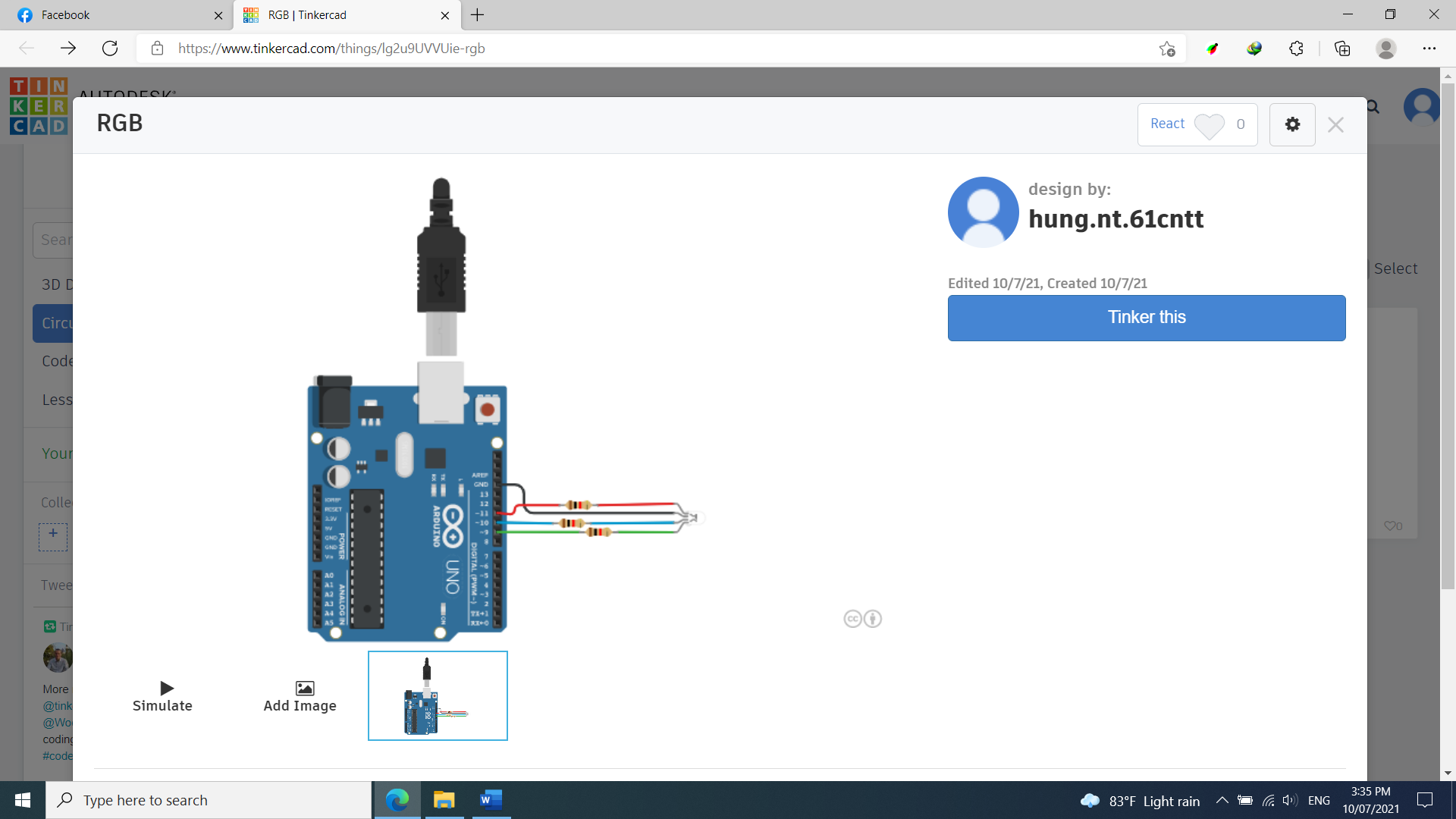
## Mô tả

Hệ thống được thiết kế điều khiển làm đèn RGB sáng ở màu đỏ, xanh lá cây, xanh dương, đèn được kết nối vào Arduino ở các cổng số 9, 10 ,11.

## Sơ đồ thiết kế



Hình 5. Sơ đồ mạch Proteus



Hình 6. Sơ đồ mạch Tinkercad

## Đặc điểm của linh kiện

* 3 điện trở
* 1 đèn RGB
* 1 mạch Arduino Uno

## Code chương trình

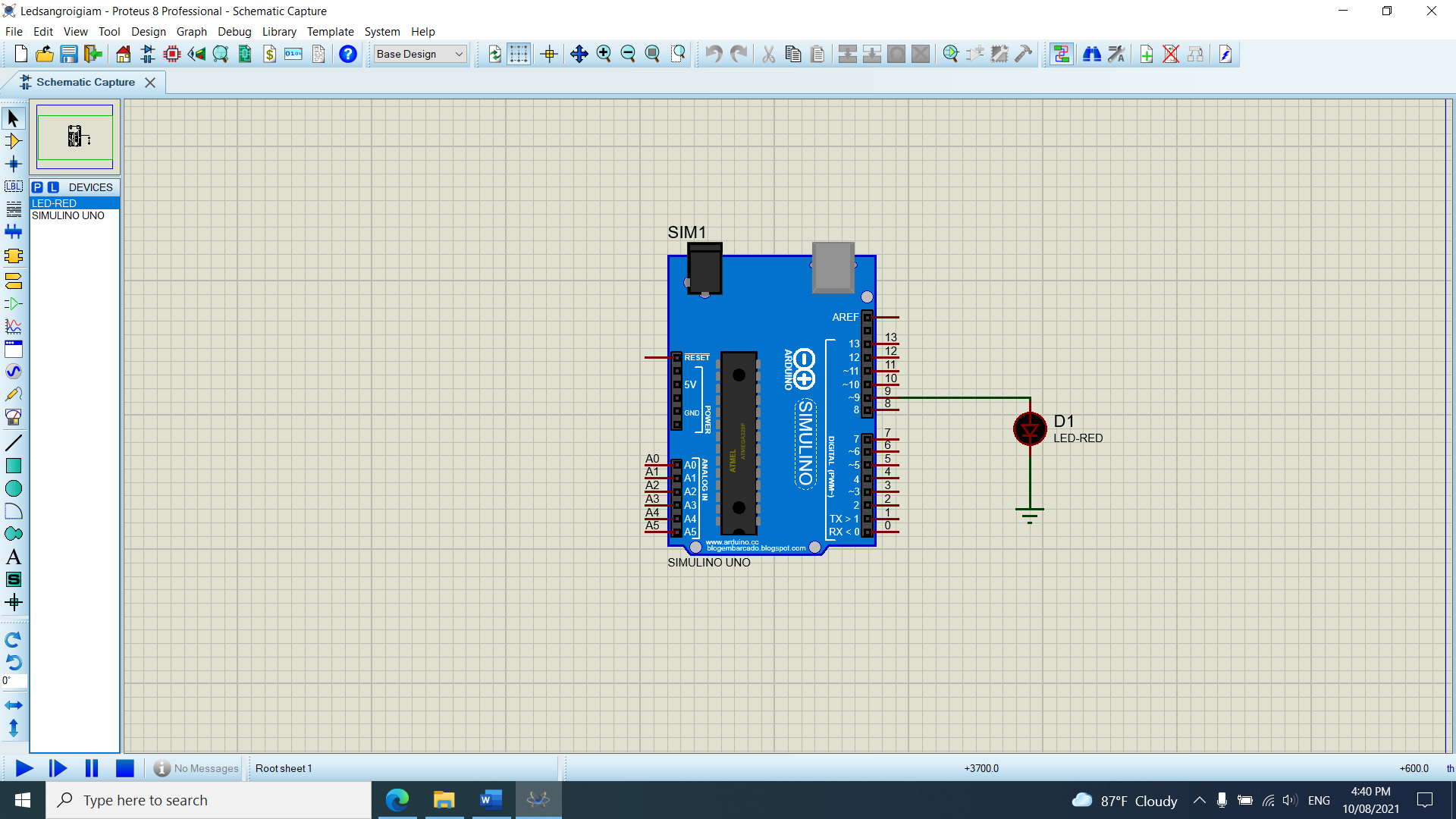
|  |
| --- |
| int denDo = 11;  int denXanh = 10;  int denXanhLa = 9;  void **setup**() {  pinMode(denDo, OUTPUT);  pinMode(denXanh, OUTPUT);  pinMode(denXanhLa, OUTPUT);  }  void **loop**() {  analogWrite(denDo, 0);  analogWrite(denXanh, 255);  analogWrite(denXanhLa, 0);  delay(500);  analogWrite(denDo, 255);  analogWrite(denXanh, 0);  analogWrite(denXanhLa, 0);  delay(500);  analogWrite(denDo, 0);  analogWrite(denXanh, 0);  analogWrite(denXanhLa, 255 ;  delay(500);  } |

# Bài 4. Đèn LED sáng dần

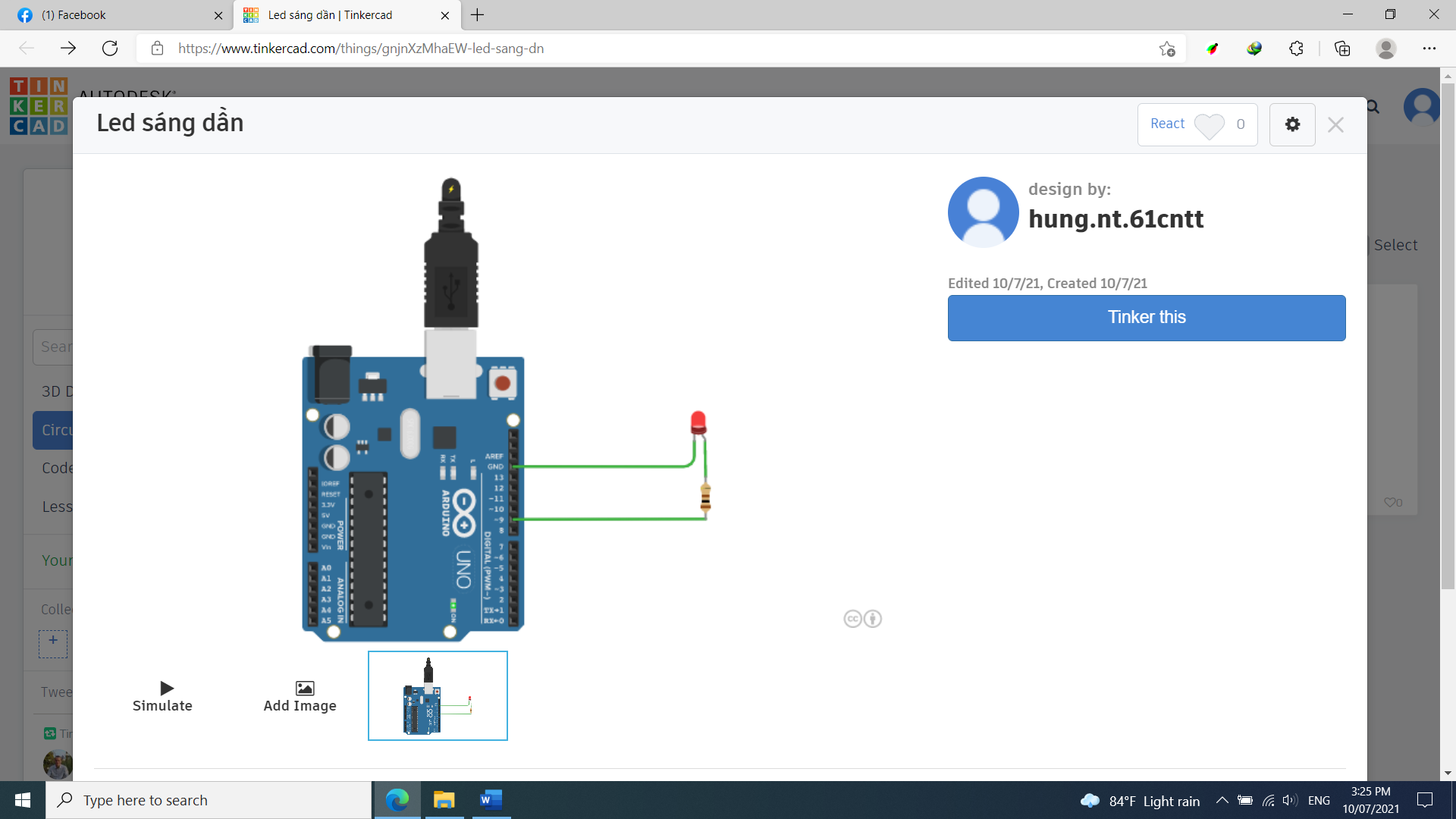
## Mô tả

Hệ thống được thiết kế Bật/ Tắt đèn Led sáng dần sau đó giảm dần, đèn được kết nối vào Arduino ở cổng số 9.

## Sơ đồ thiết kế



Hình 7. Sơ đồ mạch Proteus



Hình 8. Sơ đồ mạch Tinkercad

## Đặc điểm của linh kiện

* 1 đèn LED
* 1 Điện trở 100Ω
* 1 mạch Arduino Uno

## Code chương trình

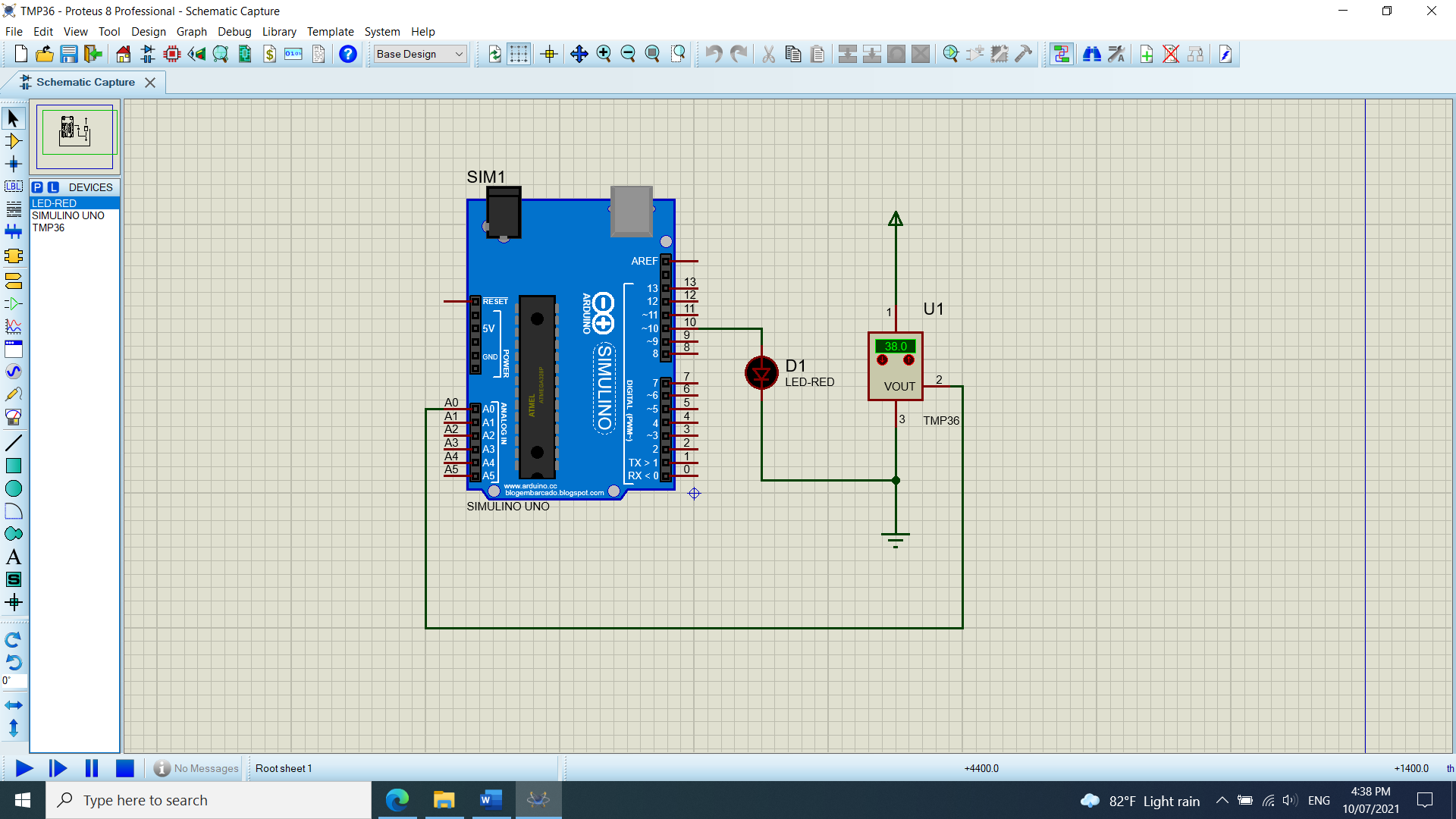
|  |
| --- |
| int i = 0;  void **setup**() {  pinMode(9,OUTPUT);  }  void **loop**() {  for (i = 0; i<=255; i +=5) {  analogWrite(9,i);  delay(30);  }  For (i = 0; i<=255; i -=5) {  analogWrite(9,i);  delay(30);  }  } |

# Bài 5. CẢM BIẾN NHIỆT ĐỘ

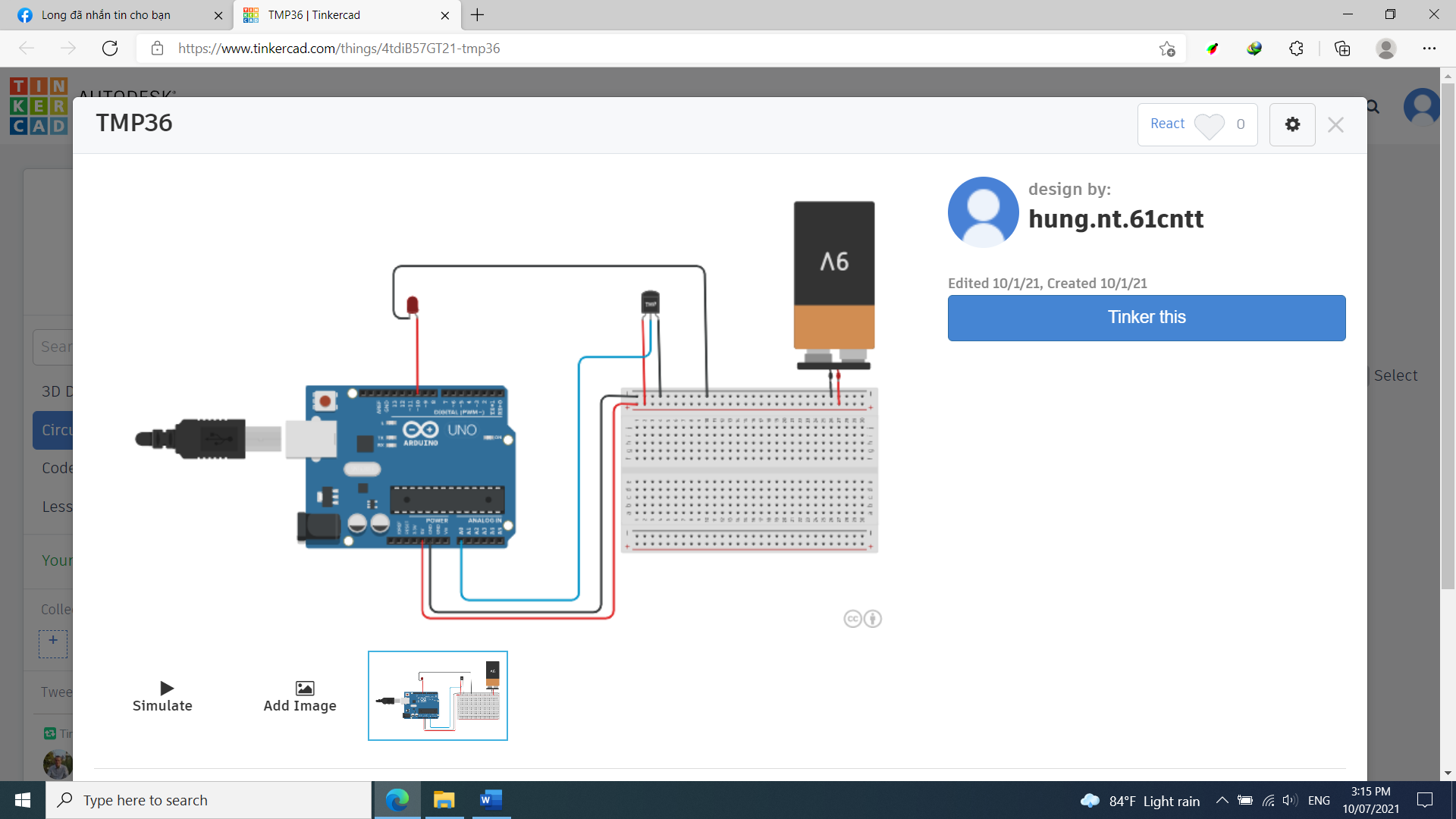
## Mô tả

Hệ thống được thiết kế thực hiện việc lập trình điều kiển khi nhiệt độ đo được trên 37 độ (tắt khi dưới 37 độ) bật đèn Led trong thời gian 1 giây, đèn Led được kết nối vào cổng số 10 của board mạch Arduino.

## Sơ đồ thiết kế



*Hình 9. Sơ đồ mạch Proteus*



*Hình 10. Sơ đồ mạch Tinkercad*

## Đặc điểm của linh kiện

* Đèn LED
* TMP36
* 1 cục pin 9V

## Code chương trình

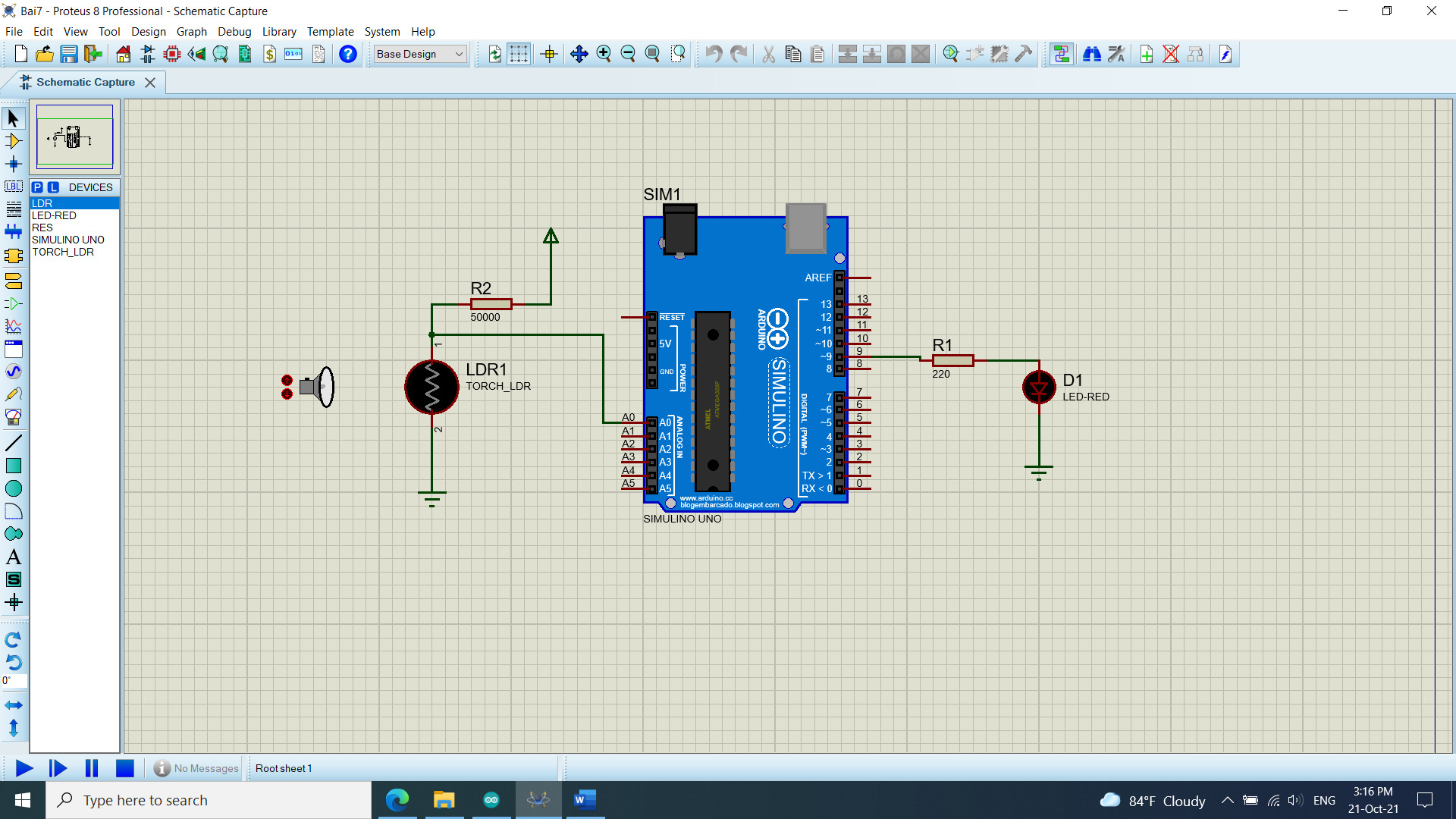
|  |
| --- |
| *void setup()*  *{*  *Serial.begin(9600);*  *}*  *void loop()*  *{*  *int gt = analogRead(A0);*  *int nd = map(gt,20,358,-40,125);*  *if(nd > 37)*  *{*  *digitalWrite(10,HIGH);*  *delay(1000);*  *}*  *else*  *{*  *digitalWrite(10,LOW);*  *delay(100);*  *}*  *}* |

# Bài 6. Điều khiển độ sáng của đèn bằng quang điện trở

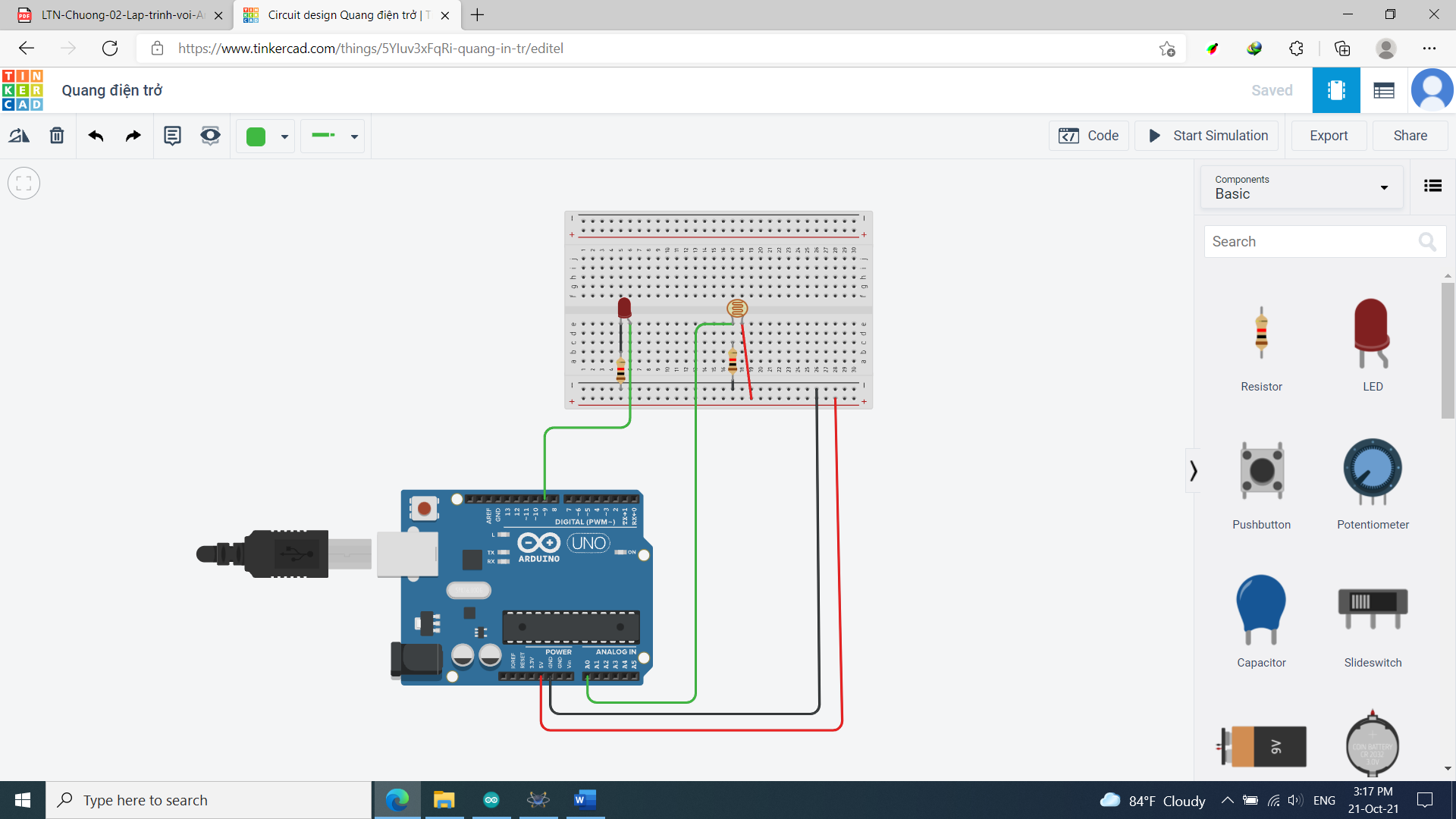
## Mô tả

Hệ thống nhúng được thiết kế để cho phép thay đổi độ sáng của Led (pin 9) thông qua một quang điện trở (gắn ở chân A0)

## Sơ đồ thiết kế



*Hình 11. Sơ đồ mạch Proteus*



*Hình 12. Sơ đồ mạch Tinkercad*

## Đặc điểm của linh kiện

* Đèn LED
* Quang điện trở
* Mạch Arduino
* 2 điện trở 220 Ω

## Code chương trình

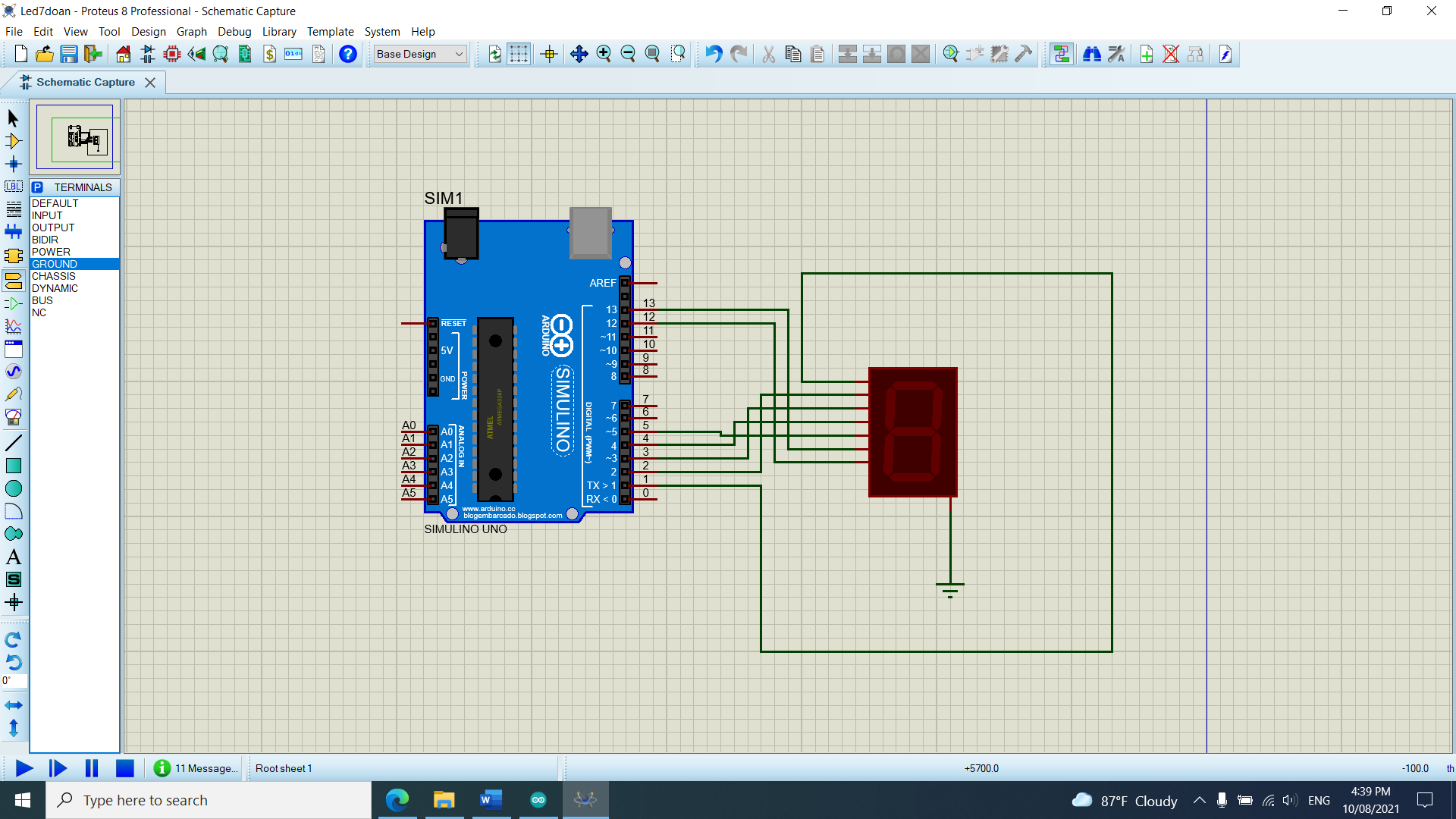
|  |
| --- |
| *int sensorValue = 0;*  *void setup()*  *{*  *pinMode(A0, INPUT);*  *pinMode(9, OUTPUT);*  *Serial.begin(9600);*  *}*  *void loop()*  *{*  *sensorValue = analogRead(A0);*  *analogWrite(9, map(sensorValue, 0, 1023, 0, 255));*  *delay(100);*  *}* |

# Bài 7. Led 7 đoạn

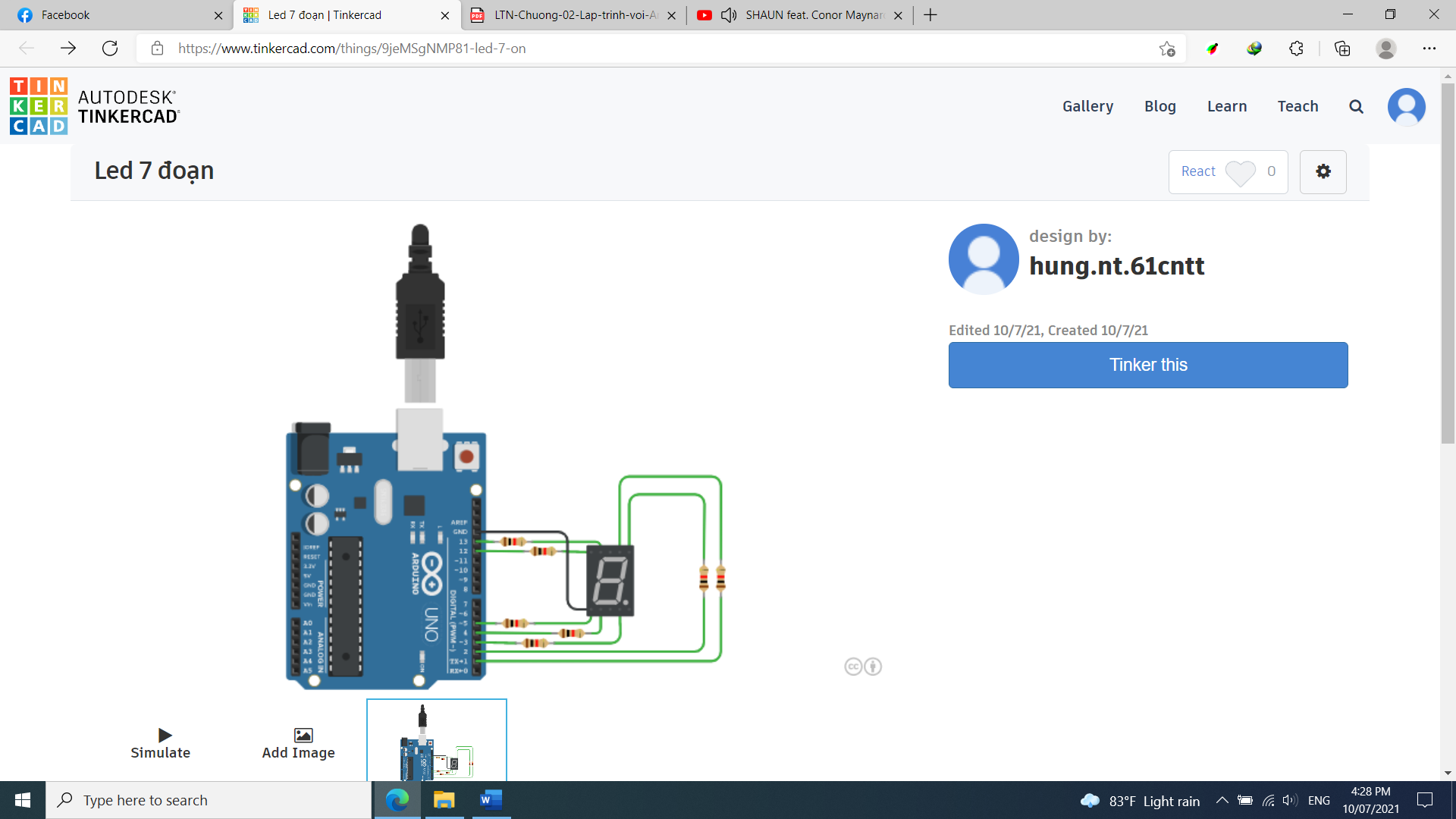
## Mô tả

Hệ thống được thiết kế thực hiện việc lập trình điều kiển bật/tắt đèn Led 7 đoạn trong thời gian 2 giây, đèn Led được kết nối vào cổng số 1,2,3,4,5,12,13 của board mạch Arduino.

## Sơ đồ thiết kế



*Hình 13. Sơ đồ mạch Protues*



*Hình 14. Sơ đồ mạch Tinkercad*

## Đặc điểm của linh kiện

* 1 đèn Led 7 màu
* 7 điện trở
* 1 mạch Arduino Uno

## Code chương trình

|  |
| --- |
| *int a=1, b=2, c=3, d=4,e=5, f=13, g=12;*  *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);*  *delay(2000);*  *}*  *void loop()*  *{*  *khong();*  *}* |

# Bài 8. Nháy lần lượt 8 led bằng IC 74HC595

## Mô tả

- Giới thiệu về IC 74595: 74HC595 là một thanh ghi dịch (shift register) hoạt động trên giao thức nối tiếp vào song song ra (Serial IN Parallel OUT).

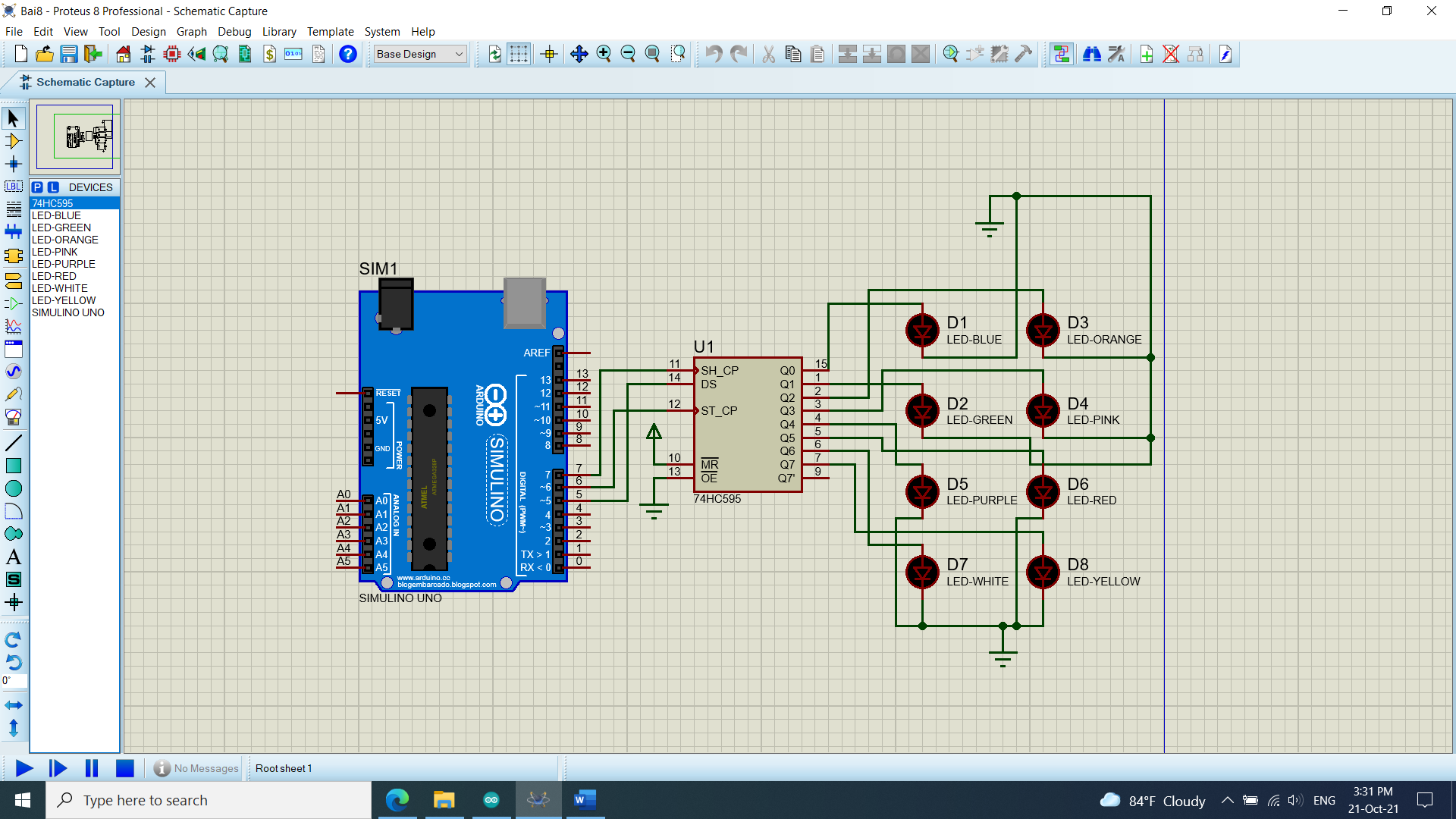
Nó nhận dữ liệu nối tiếp từ vi điều khiển và sau đó gửi dữ liệu này qua các chân song song.

Có thể tăng 8 chân đầu ra bằng cách sử dụng chip đơn.

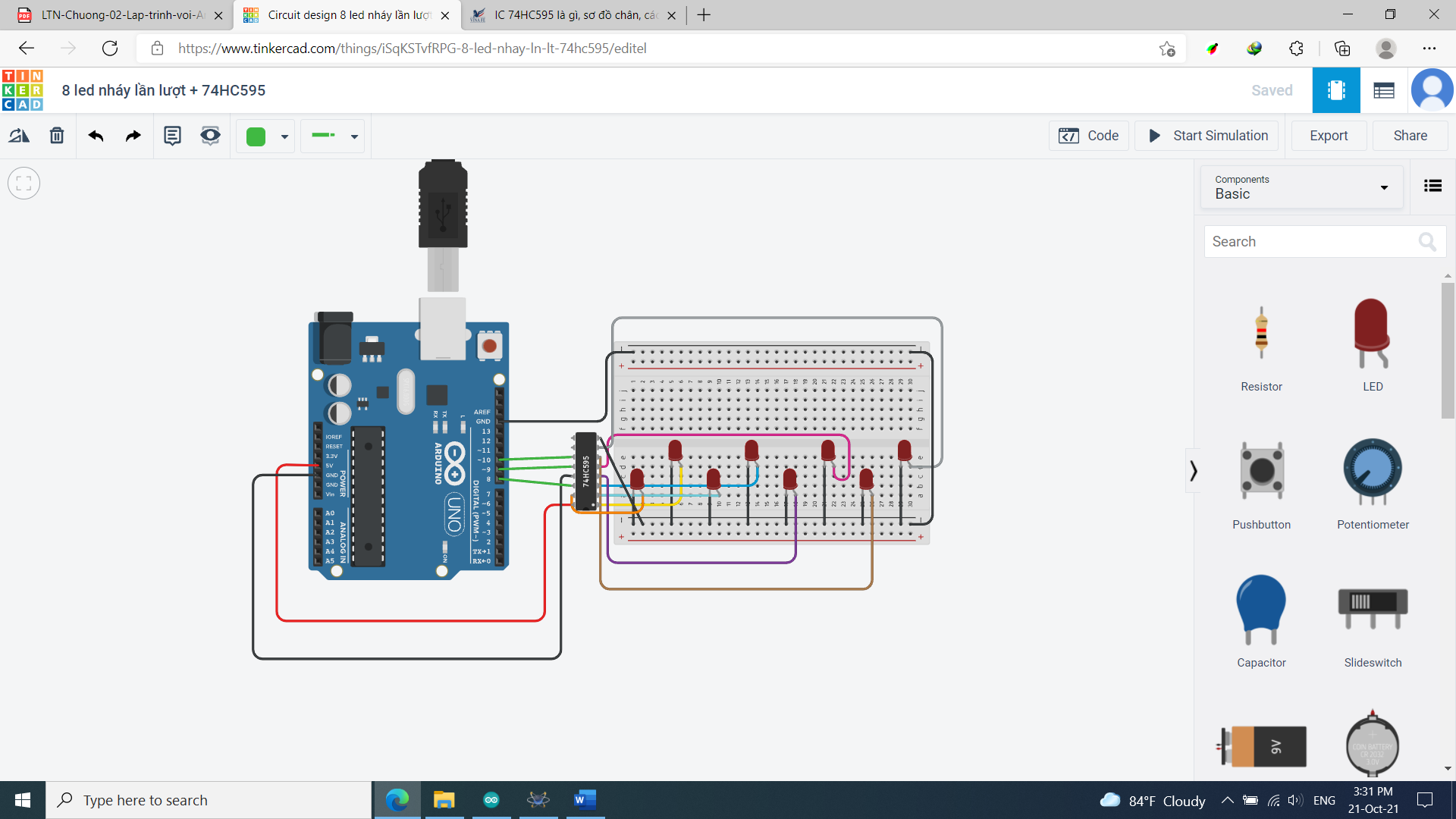
Cũng có thể kết nối song song nhiều hơn 1 thanh ghi dịch.

- Hệ thống được thiết kế thực hiện việc nháy tắt lần lượt 8 led bằng IC 74595, IC được nối với cổng A0, cổng input, latch và clock được nối vào cổng 8,9,10 của board mạch Arduino.

## Sơ đồ thiết kế



*Hình 15. Sơ đồ mạch Proteus*



*Hình 16. Sơ đồ mạch Tinkercad*

## Đặc điểm của linh kiện

* IC 74HC595
* 8 đèn led
* Mạch Arduino

## Code chương trình

|  |
| --- |
| *#define input 8*  *#define latch 9*  *#define clock 10*  *void setup()*  *{*  *pinMode(input, OUTPUT);*  *pinMode(latch, OUTPUT);*  *pinMode(clock, OUTPUT);*  *}*  *void loop()*  *{*  *for(int i=0; i<256; i++)*  *{*  *digitalWrite( latch, LOW);*  *shiftOut( input, clock, LSBFIRST, i);*  *digitalWrite( latch, HIGH);*  *delay(1000);*  *}*  *}* |

# Bài 9. Điều khiển 2 motor bằng IC L293D

## Mô tả

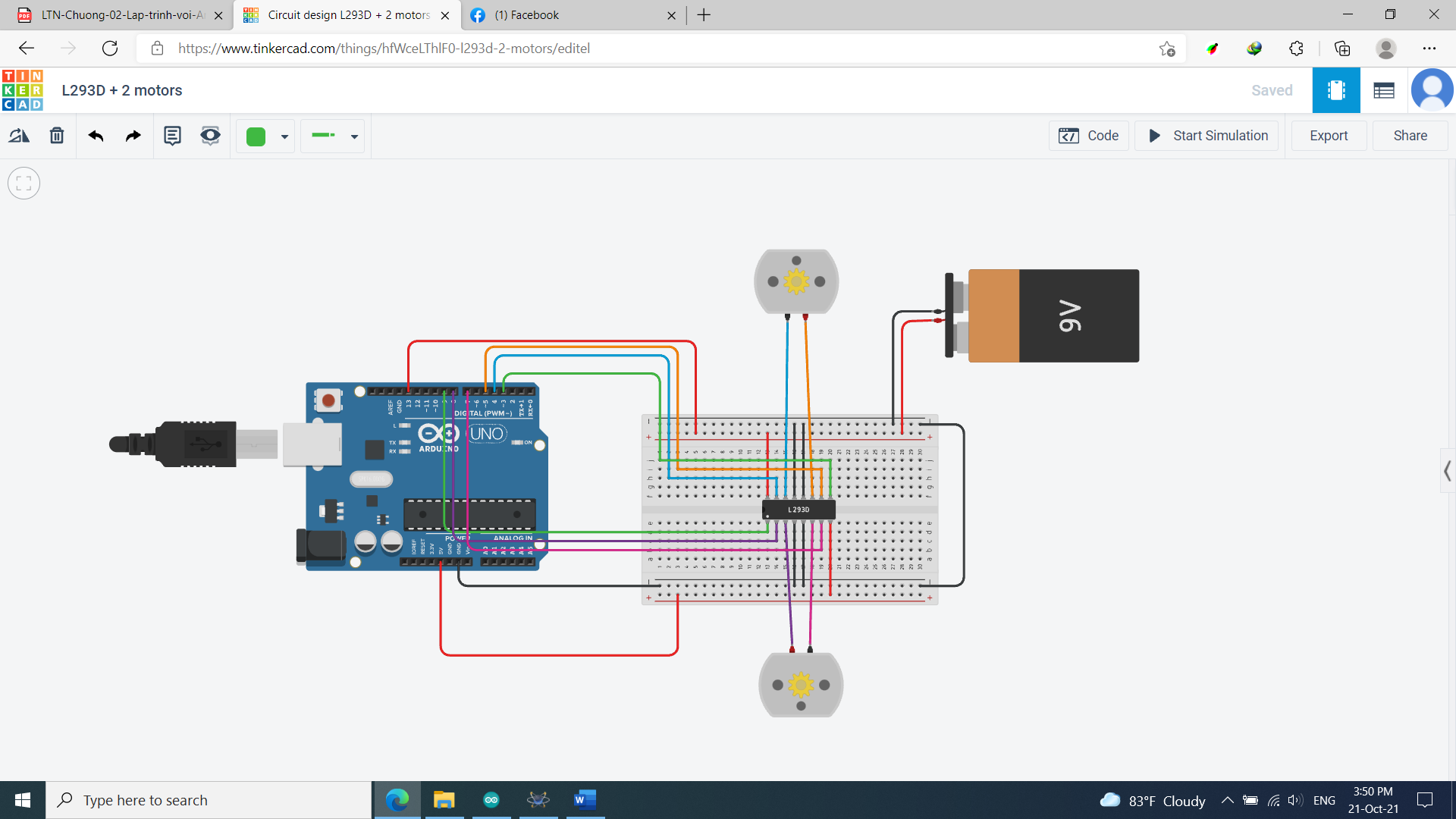
- Giới thiệu về IC L293D: L293D về cơ bản là một IC trình điều khiển hay bộ điều khiển động cơ. Nó có hai mạch cầu H tích hợp có thể điều khiển đồng thời hai động cơ DC theo cả chiều kim đồng hồ và ngược chiều kim đồng hồ. Nó hoạt động như một bộ khuếch đại dòng cao vì nó lấy tín hiệu dòng điện thấp ở đầu vào và cung cấp tín hiệu dòng điện cao hơn ở đầu ra để điều khiển các tải khác nhau, ví dụ động cơ bước và động cơ DC. Các tính năng của nó bao gồm phạm vi điện áp nguồn đầu vào lớn, tín hiệu đầu vào chống nhiễu cao dòng điện đầu ra lớn,... Các ứng dụng thực tế phổ biến của nó bao gồm trình điều khiển động cơ bước, trình điều khiển relay, trình điều khiển động cơ DC,...

- Hệ thống được thiết kế thực hiện việc điều khiển 2 motor (khởi động, tăng tốc, giảm tốc và tắt 2 motor) bằng IC L293D, cổng enA, in1, in2, enB, in3, in4 được nối lần lượt vào cổng 9,8,7,3,5,4 của board mạch Arduino.

## Sơ đồ thiết kế



*Hình 17. Sơ đồ mạch Proteus*



*Hình 18. Sơ đồ mạch Tinkercad*

## Đặc điểm của linh kiện

* IC L293D
* 2 motor
* Mạch Arduino

## Code chương trình

|  |
| --- |
| *int enA = 9;*  *int in1 = 8;*  *int in2 = 7;*  *int enB = 3;*  *int in3 = 5;*  *int in4 = 4;*  *void setup()*  *{*  *pinMode(enA, OUTPUT);*  *pinMode(in1, OUTPUT);*  *pinMode(in2, OUTPUT);*    *pinMode(enB, OUTPUT);*  *pinMode(in3, OUTPUT);*  *pinMode(in4, OUTPUT);*  *}*  *void ControlON()*  *{*  *analogWrite(enA,255);*  *analogWrite(enB,255);*    *digitalWrite(in1,HIGH);*  *digitalWrite(in2,LOW);*  *digitalWrite(in3,HIGH);*  *digitalWrite(in4,LOW);*  *}*  *void Speedup()*  *{*  *for(int i=0;i<256;i++)*  *{*  *analogWrite(enA, i);*  *analogWrite(enB, i);*  *delay(20);*  *}*  *}*  *void Speeddown()*  *{*  *for(int i=255;i>=0;--i)*  *{*  *analogWrite(enA, i);*  *analogWrite(enB, i);*  *delay(20);*  *}*  *}*  *void ControlOFF()*  *{*  *analogWrite(enA,255);*  *analogWrite(enB,255);*    *digitalWrite(in1,LOW);*  *digitalWrite(in2,LOW);*  *digitalWrite(in3,LOW);*  *digitalWrite(in4,LOW);*  *}*  *void loop()*  *{*  *ControlON();*  *delay(1000);*  *Speedup();*  *Speeddown();*  *ControlOFF();*  *delay(1000);*  *}* |

# Bài thi giữa kỳ: Đề thi 1

## Mô tả

Bao gồm nhiều mạch điện tử:

Câu 1

a) Liên tục đọc nhiệt độ môi trường và in ra màn LCD mỗi giây

b) Lập trình nút bấm để bật/ tắt đèn điện

Câu 2. Hệ thống nhúng với led 7 đoạn

a) Hiện từ 0 đến 9

b) Hiện từ 00 đến 99

Câu 3. 1 xe 3 bánh, 1 motor điều khiển bánh trước, 2 motor điều khiển 2 bánh sau, sử dụng cảm biến sóng âm HC-SR04

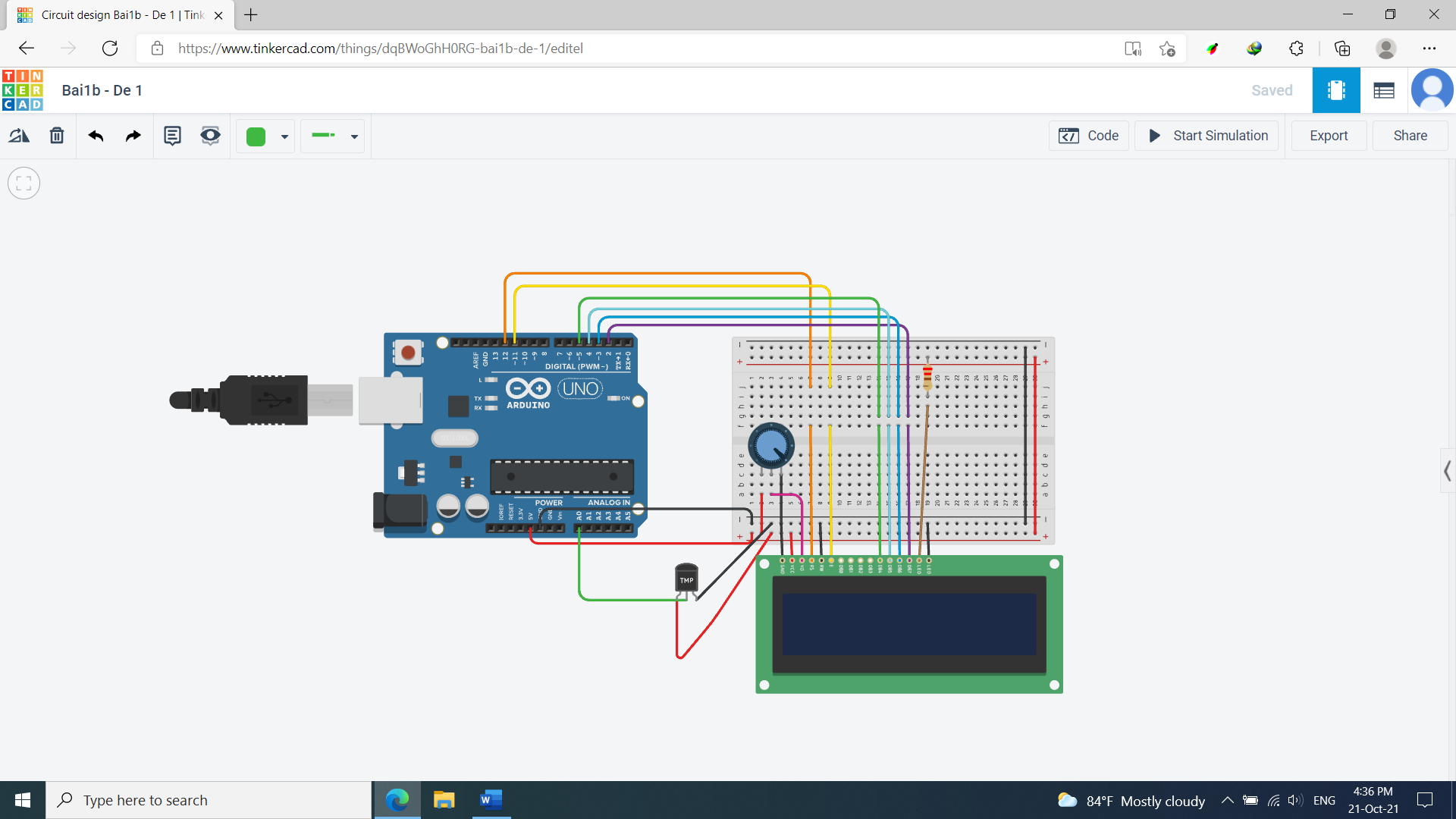
- Liên tục thực hiện phát hiện vật cản phía trước sau mỗi giây. Khi vật cản ở khoảng cách X mét thì thực hiện:

+ Nếu tốc độ xe **chậm**, đánh tay lái sang trái **10 độ** và giảm tốc độ **K** đơn vị

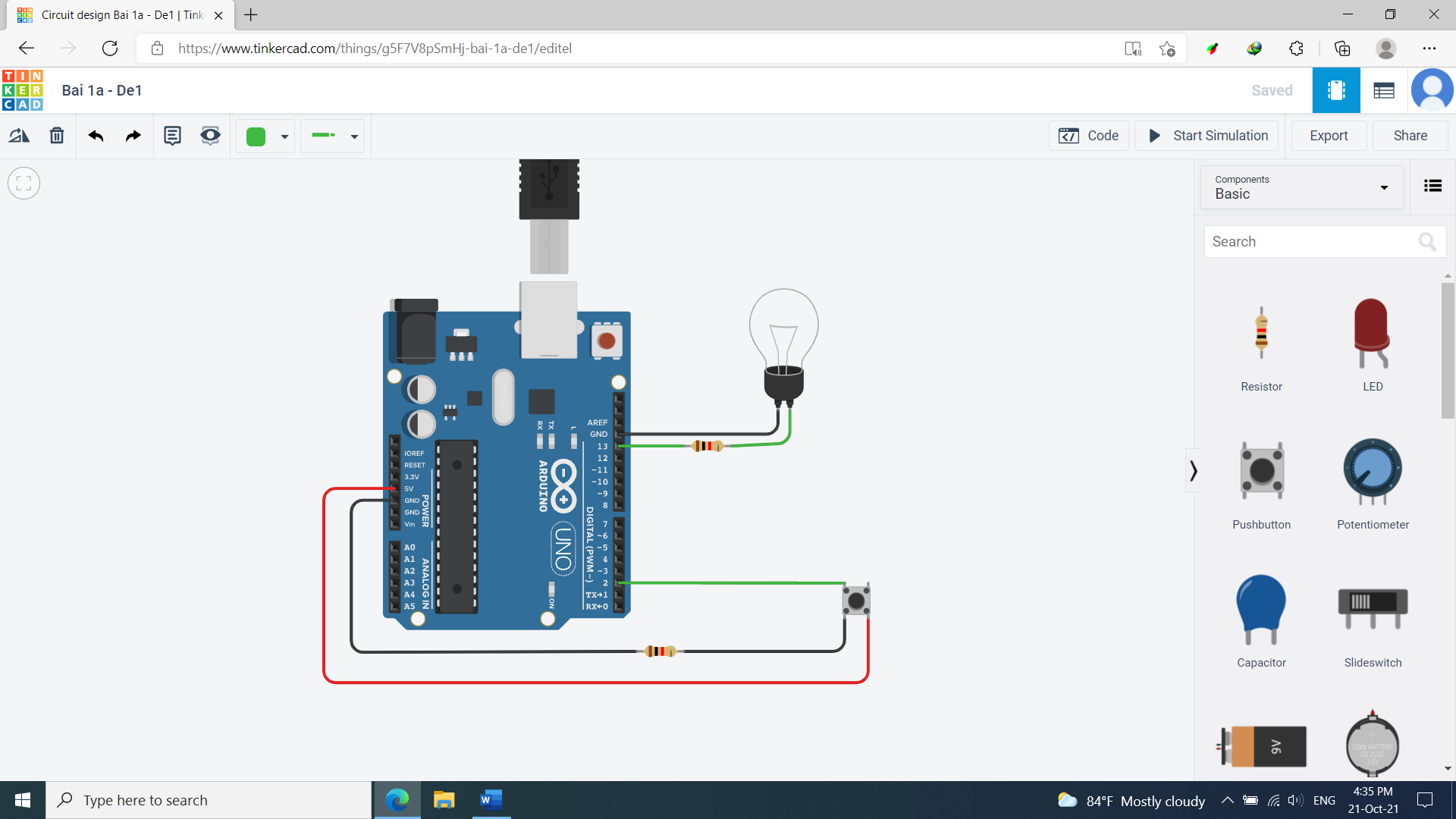
+ Nếu tốc độ xe **nhanh**, đánh tay lái sang trái **20 độ** và giảm tốc độ **K** + 10 đơn vị

+ Nếu tốc độ của xe **rất nhanh**, đánh tay lái sang trái **30 độ** và giảm tốc độ **K** + 20 đơn vị

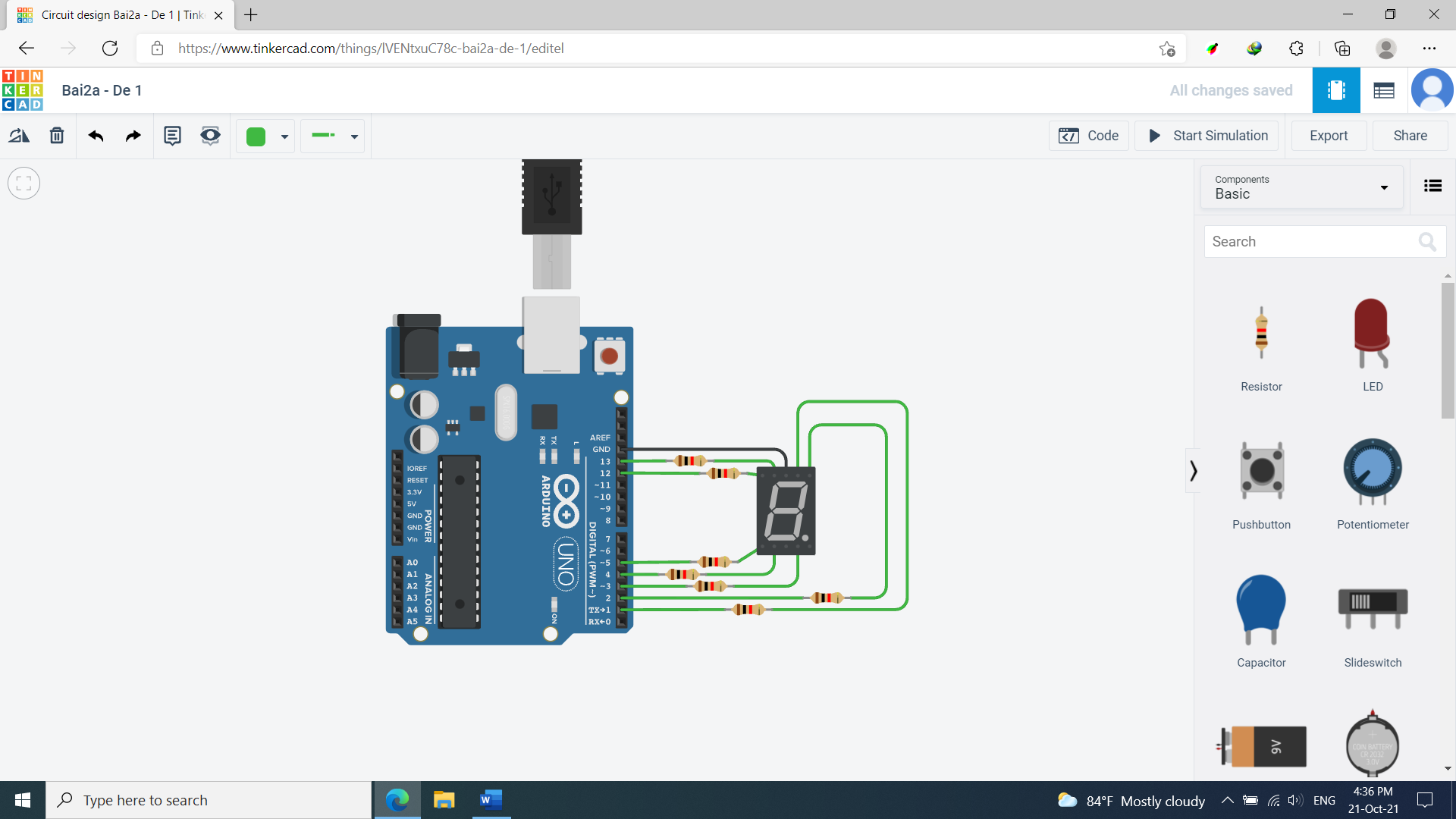
## Sơ đồ thiết kế



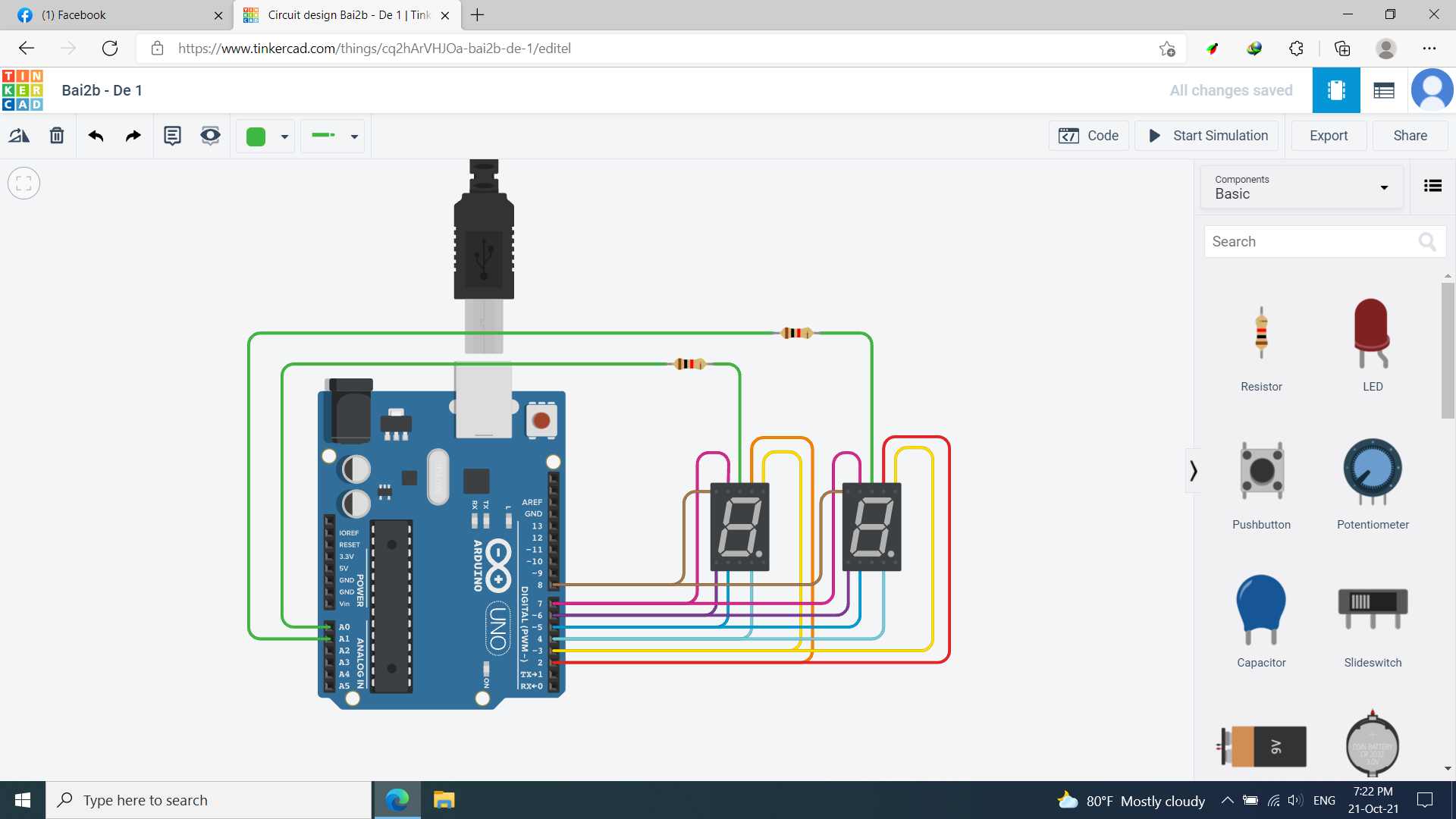
*Bài 1a*



*Bài 1b*



*Bài 2a*



*Bài 2b*

*Bài 3*

## Đặc điểm của linh kiện

**Bài 1a**

* Mạch Arduino
* 1 màn hình LCD
* TMP36

**Bài 1b**

* Mạch Arduino
* 1 bóng đèn điện
* 1 nút bấm
* 2 điện trở 100Ω

**Bài 2a**

* Mạch Arduino
* 7 điện trở
* 1 led 7 đoạn

**Bài 2b**

* Mạch Arduino
* 2 điện trở
* 2 led 7 đoạn

## Code chương trình

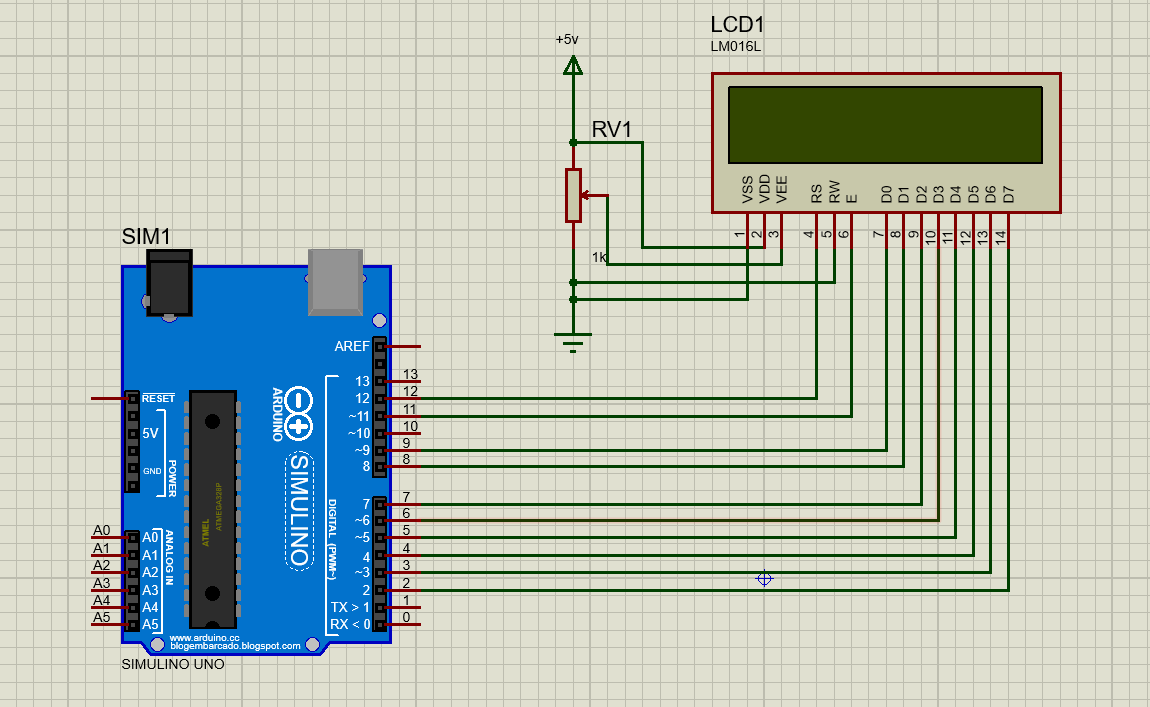
|  |
| --- |
| ***Bài 1a***  *#define light 13*  *#define btn 2*  *void setup()*  *{*  *pinMode(btn,INPUT);*  *pinMode(light, OUTPUT);*  *}*  *void loop()*  *{*  *if(digitalRead(btn)==0)*  *{*  *digitalWrite(light, LOW);*  *}*  *else*  *{*  *digitalWrite(light, HIGH);*  *}*  *}*  ***Bài 1b***  *#include <LiquidCrystal.h>*  *LiquidCrystal lcd(12, 11, 5, 4, 3, 2);*  *void setup() {*  *lcd.begin(16, 2);*  *lcd.print("Nhiet do do duoc");*  *Serial.begin(9600);*  *}*  *void loop() {*  *lcd.setCursor(0, 1);*  *int gt = analogRead(A0);*  *int nd = map(gt,20,358,-40,125);*  *if(nd > 37)*  *{*  *digitalWrite(10,HIGH);*  *delay(1000);*  *}*  *else*  *{*  *digitalWrite(10,LOW);*  *delay(100);*  *}*  *lcd.print(nd);*  *}*  ***Bai 2a***  *int a=1, b=2, c=3, d=4,e=5, f=13, g=12;*  *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);*  *delay(2000);*  *}*  *void mot()*  *{*  *digitalWrite(a, LOW); digitalWrite(b, HIGH);*  *digitalWrite(c, HIGH); digitalWrite(d, LOW);*  *digitalWrite(e, LOW); digitalWrite(f, LOW);*  *digitalWrite(g, LOW);*  *delay(2000);*  *}*  *void hai()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, HIGH);*  *digitalWrite(c, LOW); digitalWrite(d, HIGH);*  *digitalWrite(e, HIGH); digitalWrite(f, LOW);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void ba()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, HIGH);*  *digitalWrite(c, HIGH); digitalWrite(d, HIGH);*  *digitalWrite(e, LOW); digitalWrite(f, LOW);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void bon()*  *{*  *digitalWrite(a, LOW); digitalWrite(b, HIGH);*  *digitalWrite(c, HIGH); digitalWrite(d, LOW);*  *digitalWrite(e, LOW); digitalWrite(f, HIGH);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void nam()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, LOW);*  *digitalWrite(c, HIGH); digitalWrite(d, HIGH);*  *digitalWrite(e, LOW); digitalWrite(f, HIGH);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void sau()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, LOW);*  *digitalWrite(c, HIGH); digitalWrite(d, HIGH);*  *digitalWrite(e, HIGH); digitalWrite(f, HIGH);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void bay()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, HIGH);*  *digitalWrite(c, HIGH); digitalWrite(d, LOW);*  *digitalWrite(e, LOW); digitalWrite(f, LOW);*  *digitalWrite(g, LOW);*  *delay(2000);*  *}*  *void tam()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, HIGH);*  *digitalWrite(c, HIGH); digitalWrite(d, HIGH);*  *digitalWrite(e, HIGH); digitalWrite(f, HIGH);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void chin()*  *{*  *digitalWrite(a, HIGH); digitalWrite(b, HIGH);*  *digitalWrite(c, HIGH); digitalWrite(d, HIGH);*  *digitalWrite(e, LOW); digitalWrite(f, HIGH);*  *digitalWrite(g, HIGH);*  *delay(2000);*  *}*  *void loop()*  *{*  *khong();*  *mot();*  *hai();*  *ba();*  *bon();*  *nam();*  *sau();*  *bay();*  *tam();*  *chin();*  *}*  ***Bài 2b***  *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}; //{a,b,c,d,e,f,g}*  *byte digitPin[2] = {A0,A1}; //segment*  *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()*  *{*  *for (int i = 0; i<100; i++)*  *{*  *display\_N(i);*  *delay(1);*  *}*  *}*  *void display\_N(int num)*  *{*  *int und = num % 10;*  *int dec = (num % 100) / 10;*  *for(int i=0; i<100; i++)*  *{*  *segOutput(1, und, 1);*  *segOutput(0, dec, 1);*  *delay(2);*  *}*    *Serial.print(dec);*  *Serial.println(und);*  *}*    *//LED*  *void segClear(){*    *for(int i=0;i<8;i++){*  *digitalWrite(segPin[i], HIGH);*  *}*  *}*  *//LED*  *void segOutput(int d, int Number, int dp){*    *segClear();*  *digitalWrite(digitPin[d], HIGH);*    *for(int i=0;i<8;i++){*  *digitalWrite(segPin[i], segValue[Number][i]);*  *}*    *digitalWrite(segPin[7], dp);*    *delayMicroseconds(500);*    *digitalWrite(digitPin[d], LOW);*    *}* |

# Bài 10. Điều khiển hiển thị trên màn hình LCD bằng Arduino

## Mô tả

Hệ thống lập trình điều khiển hiển thì lên trên màn hình LCD bằng mạch Arduino và thư viện LiquidCrystal

## Sơ đồ thiết kế



Hình 17. Sơ đồ mạch Proteus

## Đặc điểm của linh kiện

* Mạch Arduino
* 1 điện trở 100Ω
* 1 màn hình LCD

## Code chương trình

|  |
| --- |
| *#include<LiquidCrystal.h>*  *LiquidCrystal lcd(12,11,9,8,7,6,5,4,3,2);*  *void setup()*  *{*  *lcd.begin(16,2);*  *lcd.print("Hello World!");*  *}*  *void loop()*  *{*  *lcd.setCursor(0,1);*  *lcd.print("61CNTT-2");*  *}* |

# Bài 11.Hiển thị led hình trái tim bằng mạch STM32F401VE

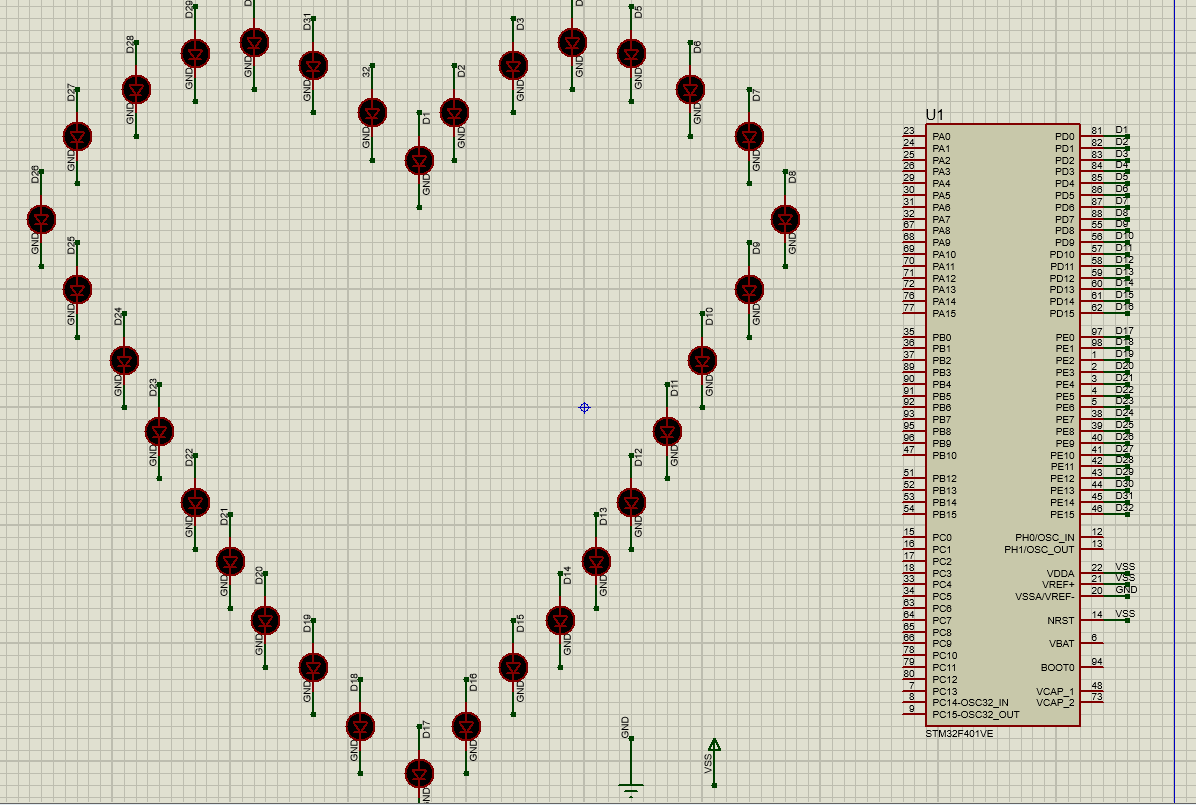
## Mô tả

Tập đoàn ST Microelectronic cho ra mắt dòng STM32 , vi điều khiển đầu tiên dựa trên nền lõi ARM Cortex - M 3 thế hệ mới do hãng ARM thiết kế, lõi ARM Cortex - M 3 là sự cải tiến của lõi ARM 7 truyền thống

Dòng ARM Cortex ™ - M là thế hệ mới, thiế t lập các tiêu chuẩn mới về hiệu suất, chi phí, ứng dụng cho các thiết bị cần tiêu thụ năng lượng thấp, và đáp ứng yêu cầu thời gian thực khắ t khe

Mạch lập trình hiển thị led hình trái tim

## Sơ đồ thiết kế



Hình 18. Sơ đồ mạch Arduino

## Đặc điểm của linh kiện

* Mạch STM32F401VE
* 33 led

## Code chương trình

|  |
| --- |
| *#include "main.h"*  *void SystemClock\_Config(void);*  *static void MX\_GPIO\_Init(void);*  *void Sanghetled();*  *void Tathetled();*  *int main(void)*  *{*  *HAL\_Init();*  *SystemClock\_Config();*  *MX\_GPIO\_Init();*  *while (1)*  *{*  *Sanghetled();*  *HAL\_Delay(1000);*  *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);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_8, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_9, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_10, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_11, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_12, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_13, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_14, GPIO\_PIN\_SET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_15, 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);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_8, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_9, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_10, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_11, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_12, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_13, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_14, GPIO\_PIN\_RESET);*  *HAL\_GPIO\_WritePin(GPIOE, GPIO\_PIN\_15, GPIO\_PIN\_RESET);*  *}* |

# Bài 12.

## Mô tả

## Sơ đồ thiết kế

## Đặc điểm của linh kiện

## Code chương trình

# Bài 11.

## Mô tả

## Sơ đồ thiết kế

## Đặc điểm của linh kiện

## Code chương trình

# Bài 11.

## Mô tả

## Sơ đồ thiết kế

## Đặc điểm của linh kiện

## Code chương trình