

***Asignatura:***

Programación de sistemas

***Sustentado por:***

Carlos David Cerrato Díaz – 61841055

Nelson Enrique Cerrato Ramos – 62011406

Luis Antonio Suazo Bonilla – 61751038

Wuesley Gómez – 61811148

Gabriel Montoya – 61911582

Chelsea Massiell Suazo García – 61911443

**Dirigido:**

Ing. Kevin Cruz

***Asignación:***

2do Avance de proyecto

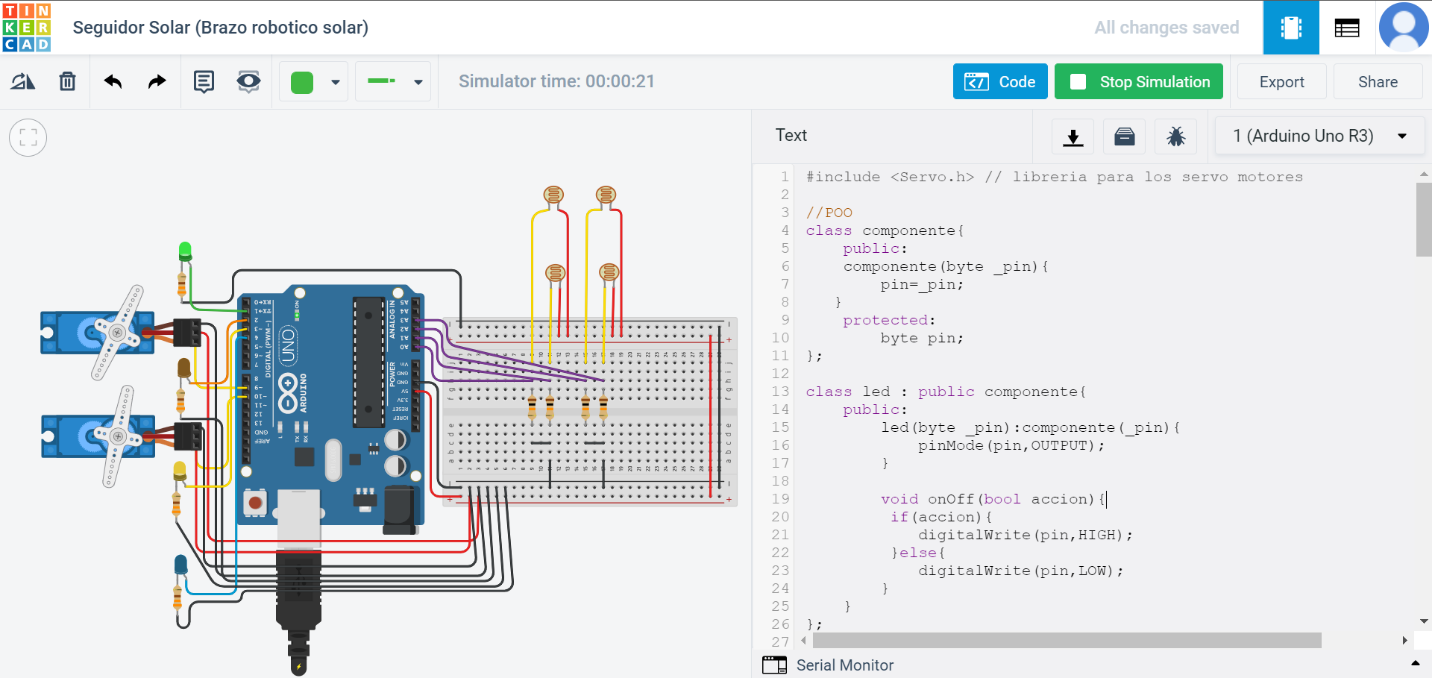
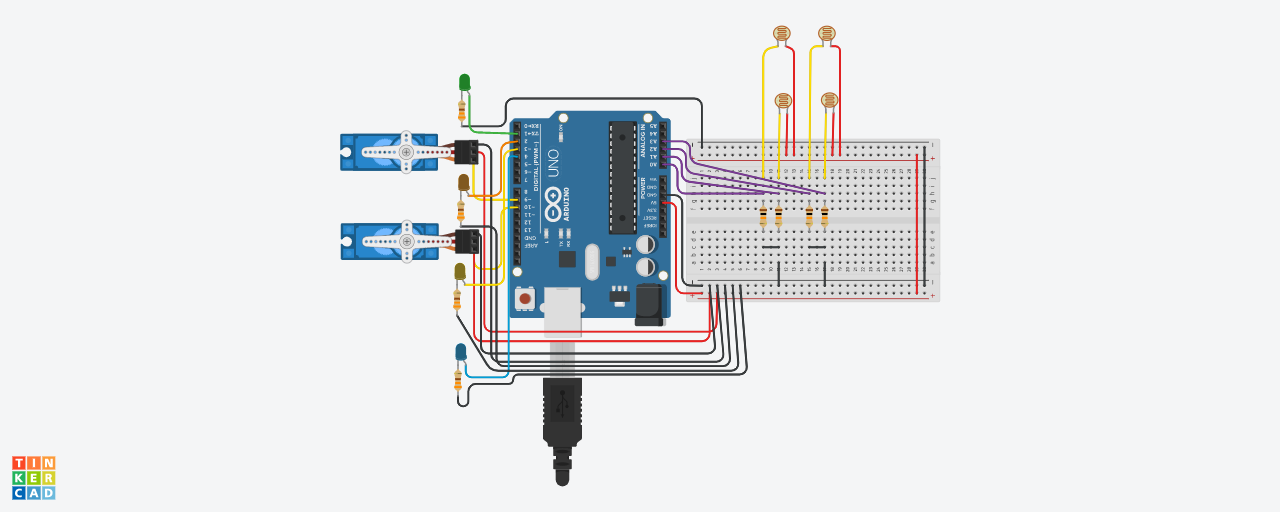
***Campus:***

Ceutec Sede Norte

***Fecha:***

07/06/2021

**SEGUIDOR SOLAR (BRAZO ROBÓTICO SOLAR)**



#include <Servo.h> // libreria para los servo motores

//POO

class componente{

public:

componente(byte \_pin){

pin=\_pin;

}

protected:

byte pin;

};

class led : public componente{

public:

led(byte \_pin):componente(\_pin){

pinMode(pin,OUTPUT);

}

void onOff(bool accion){

if(accion){

digitalWrite(pin,HIGH);

}else{

digitalWrite(pin,LOW);

}

}

};

//Objetos de clase creada (LED)

led luzVerde(1);

led luzNaranja(2);

led luzAmarillo(3);

led luzAzul(4);

// 180 horizontal MAXIMO

Servo horizontal; // declaramos el valor para el servo horizontal

int servoh = 180;

int servohLimitHigh = 180; //limite en posicion superior

int servohLimitLow = 65; //limite para posicion inferior

Servo vertical; // vertical servo

int servov = 45;

int servovLimitHigh = 80;

int servovLimitLow = 15;

int ldrlt = 0; // conexion a las resistencias LDR

int ldrrt = 1;

int ldrld = 2;

int ldrrd = 3;

void setup()

{

Serial.begin(9600);

horizontal.attach(9); //servo horizontal pin digital 9

vertical.attach(10); // servo vertical pin digital 10

horizontal.write(180);//arrancamos programa, motor se inicializa en 180

vertical.write(45);//arrancamos programa, motor se inicializa en 45

delay(3000);

}

void loop()

{

int lt = analogRead(ldrlt); // top left (superior izquierdo)

int rt = analogRead(ldrrt); // top right (superior derecho)

int ld = analogRead(ldrld); // down left (inferior izquierdo)

int rd = analogRead(ldrrd); // down rigt (inferior derecho)

int avt = (lt + rt) / 2; // valor superior maximo

int avd = (ld + rd) / 2; // valor inferior maximo

int avl = (lt + ld) / 2; // valor izquierdo maximo

int avr = (rt + rd) / 2; // valor derecho maximo

int dvert = avt - avd; // diferencia entre superior e inferior

int dhoriz = avl - avr;// diferencia entre izquierdo y derecho

Serial.print(avt); // Imnprimir valores de posicion en monitor serial

Serial.print(" ");

Serial.print(avd);

Serial.print(" ");

Serial.print(avl);

Serial.print(" ");

Serial.print(avr);

Serial.print(" ");

if (avt > avd)

{

servov = ++servov;

if (servov > servovLimitHigh)

{

servov = servovLimitHigh;

luzAmarillo.onOff(1);

delay(2000);

luzAzul.onOff(0);

delay(2000);

}

}

else if (avt < avd)

{

servov= --servov;

if (servov < servovLimitLow)

{

servov = servovLimitLow;

luzAmarillo.onOff(0);

delay(2000);

luzAzul.onOff(1);

delay(2000);

}

}

vertical.write(servov);

if (avl > avr)

{

servoh = --servoh;

if (servoh < servohLimitLow)

{

servoh = servohLimitLow;

luzVerde.onOff(1);

delay(2000);

luzNaranja.onOff(0);

delay(2000);

}

}

else if (avl < avr)

{

servoh = ++servoh;

if (servoh > servohLimitHigh)

{

servoh = servohLimitHigh;

luzVerde.onOff(0);

delay(2000);

luzNaranja.onOff(1);

delay(2000);

}

}

else if (avl = avr)

{

}

horizontal.write(servoh);

}