

	Computación	Docente: Diego Quisi Peralta
	Programación Aplicada	Período Lectivo: Septiembre 2020 – Febrero 2021

		FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES	
CARRERA: COMPUTACIÓN/INGENIERÍA DE SISTEMAS		ASIGNATURA: PROGRAMACIÓN APLICADA	
NRO. PROYECTO:	1.1	TÍTULO PROYECTO: Practica de Arduino Desarrollo e implementación de un entorno de simulación Online para placas Arduino y electrónica básica.	
OBJETIVO: Reforzar los conocimientos adquiridos en clase sobre Arduino.			
INSTRUCCIONES:		1. Revisar el contenido teórico y práctico del tema.	
		2. Profundizar los conocimientos revisando los libros guías, los enlaces contenidos en los objetos de aprendizaje Java y la documentación disponible en fuentes académicas en línea de Arduino.	
		3. Crear una cuenta dentro de la herramienta Online https://www.tinkercad.com para simular circuitos eléctricos.	
		4. Revisar los siguientes videos que le ayudaran para realizar la tarea: - https://www.youtube.com/watch?v=r25dG32IWSU (Video de Electrónica Básica) - https://www.youtube.com/watch?v=hZmSG-IALAM (Video de Arduino Básico)	
		5. Revisar el ejemplo subido al AVAC del prender un led dentro del simulador TinkerCad y cargar en la herramienta para ver la simulación (PrenderApagarLed.brd). 6. Subir el informe de la practica en formato PDF y los archivos al Git Personal. Fecha de Entrega: 31 de enero 2021	
ACTIVIDADES POR DESARROLLAR			

	Computación	Docente: Diego Quisi Peralta
	Programación Aplicada	Período Lectivo: Septiembre 2020 – Febrero 2021

1. Investigue, diseñe y desarrolle e implemente tres sistemas de simulación electrónica de Arduino dentro de la herramienta online Tinkercad.

DEFINICIÓN DEL PROBLEMA:

Tinkercad es un software gratuito para el diseño 3D desarrollado por Autodesk. En su apartado circuits ofrece un simulador online de Arduino bastante completo y fácil de utilizar.

Tinkercad ofrece bastantes componentes para armar nuestros esquemas y circuitos, y muchos de ellos se pueden configurar (como por ejemplo las resistencias y los diodos) y manipular en tiempo real (potenciómetros, botones, etc.).

La programación en TinkerCad se puede realizar en modo código y en modo bloques, y también tenemos disponible una pantalla dividida donde vemos los dos modos simultáneamente.

Al compilar el código, si hay algún error es marcado por el depurador. Una vez que tenemos el código arduino listo y la simulación funciona sin errores podemos descargar el archivo .ino para subirlo a nuestro arduino. El software nos provee de un monitor serie con plotter serial incluido similar al IDE de Arduino. TinkerCad es una plataforma ideal para quienes están aprendiendo Arduino y programación. Es muy intuitiva y de fácil manejo, gratuita y online.

En base a ello se propone resolver tres problemas electrónicos:

1. Generar un auto fantástico que se prenda y se apague desde un pulsante.
2. Generar una lámpara de ciudad, es decir que se prenda cuando es noche y se apague cuando ya exista luz para esto deben utilizar un LDR y un LED.
3. Finalmente, controlar un servomotor con un potenciómetro el grado de giro.

RESULTADO(S) OBTENIDO(S):

- Interpreta de forma correcta la programación en Arduino.
- Identifica correctamente qué herramientas de electrónica se pueden aplicar.

CONCLUSIONES:

- Los estudiantes implementan soluciones de hardware en sistemas.
- Los estudiantes están en la capacidad de implementar sistemas electrónicos en Arduino.

RECOMENDACIONES:


- Revisar la información proporcionada por el docente previo a la práctica.
- Haber asistido a las sesiones de clase.
- **Consultar con el docente las dudas que puedan surgir al momento de realizar la práctica.**

BIBLIOGRAFIA:

[1]: <https://www.ups.edu.ec/evento?calendarBookingId=98892>

Docente / Técnico Docente: Ing. Diego Quisi Peralta Msc.

Firma: _____

	Computación	Docente: Diego Quisi Peralta
	Programación Aplicada	Período Lectivo: Septiembre 2020 – Febrero 2021



FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA ESTUDIANTES

CARRERA:		ASIGNATURA:
NRO. PRÁCTICA:		TÍTULO PRÁCTICA:
OBJETIVO ALCANZADO: <ul style="list-style-type: none"> Reforzar los conocimientos adquiridos en clase sobre Arduino. 		

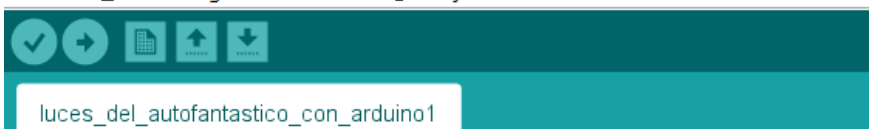
ACTIVIDADES DESARROLLADAS

1. Generar un auto fantástico que se prenda y se apague desde un pulsante.

CODIGO:

📄 `luces_del_autofantastico_con_arduino1` Arduino 1.8.13 (Windows Store 1.8.42.0)

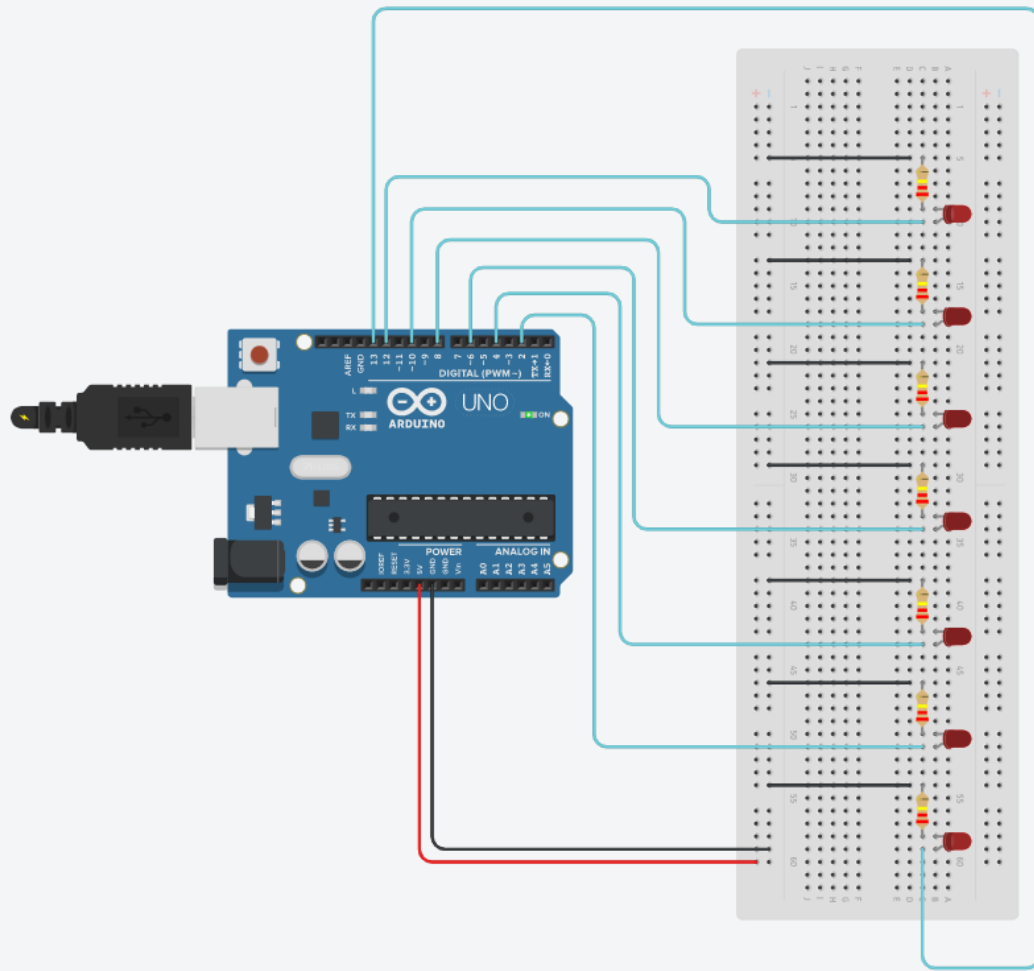
Archivo Editar Programa Herramientas Ayuda



```
void setup()
{
  pinMode(12, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(2, OUTPUT);
  pinMode(13, OUTPUT);
}

void loop()
{
  digitalWrite(12, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(12, LOW);
  digitalWrite(10, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(10, LOW);
  digitalWrite(8, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(8, LOW);
  digitalWrite(6, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(6, LOW);
  digitalWrite(4, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(4, LOW);
  digitalWrite(2, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(2, LOW);
  digitalWrite(13, HIGH);
  delay(200); // Wait for 200 millisecond(s)
  digitalWrite(13, LOW);
}
```

DEMOSTRACION DEL RESULTADO:



2. Generar una lampara de ciudad, es decir que se prenda cuando es noche y se apague cuando ya exista luz para esto deben utilizar un LDR y un LED.

CODIGO:

shiny_juttuli_fyyran1 Arduino 1.8.13 (Windows Store 1.8.42.0)

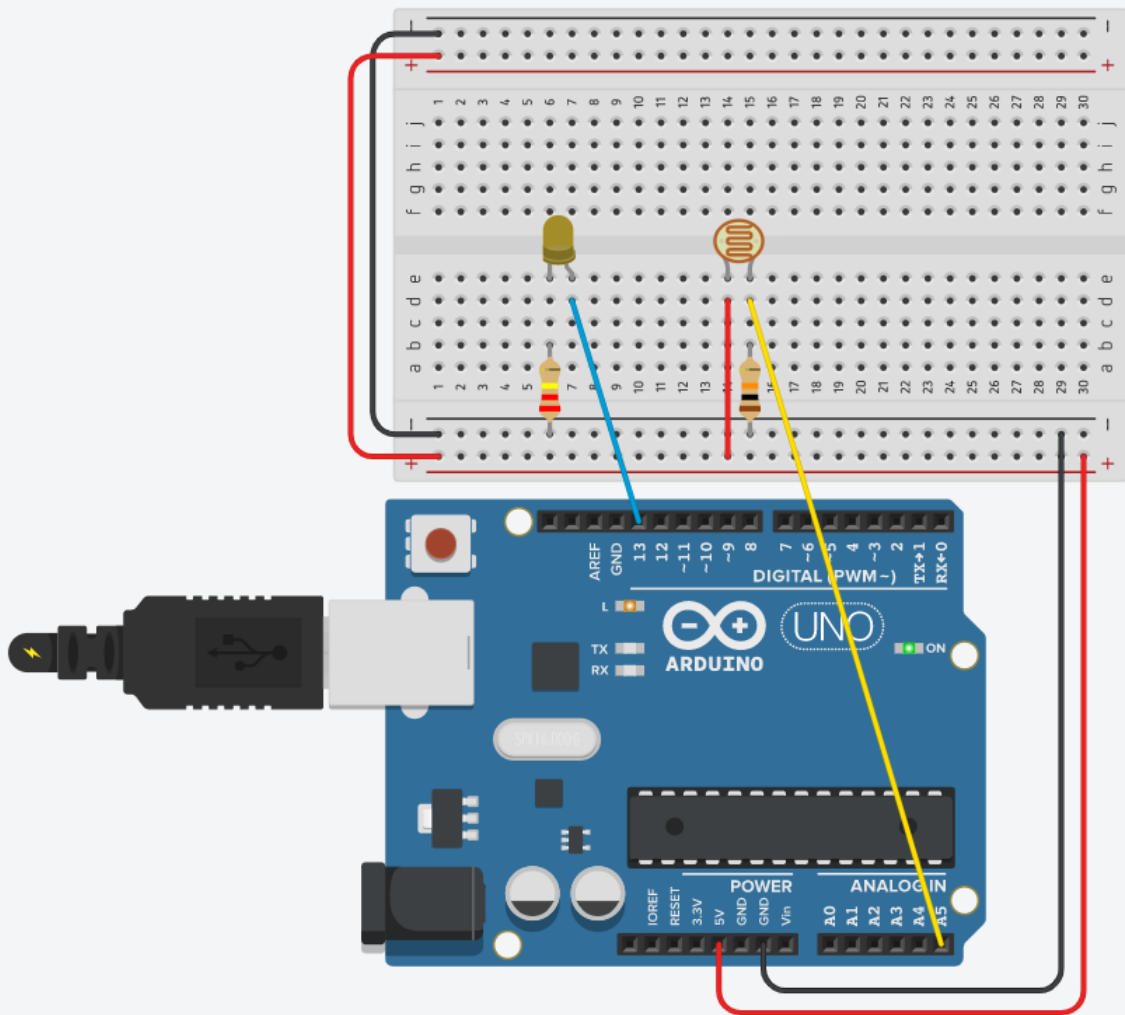
Archivo Editar Programa Herramientas Ayuda

```
shiny_juttuli_fyyran1
#define LED 13
#define LDR A5

void setup()
{
  pinMode(LED, OUTPUT);
}

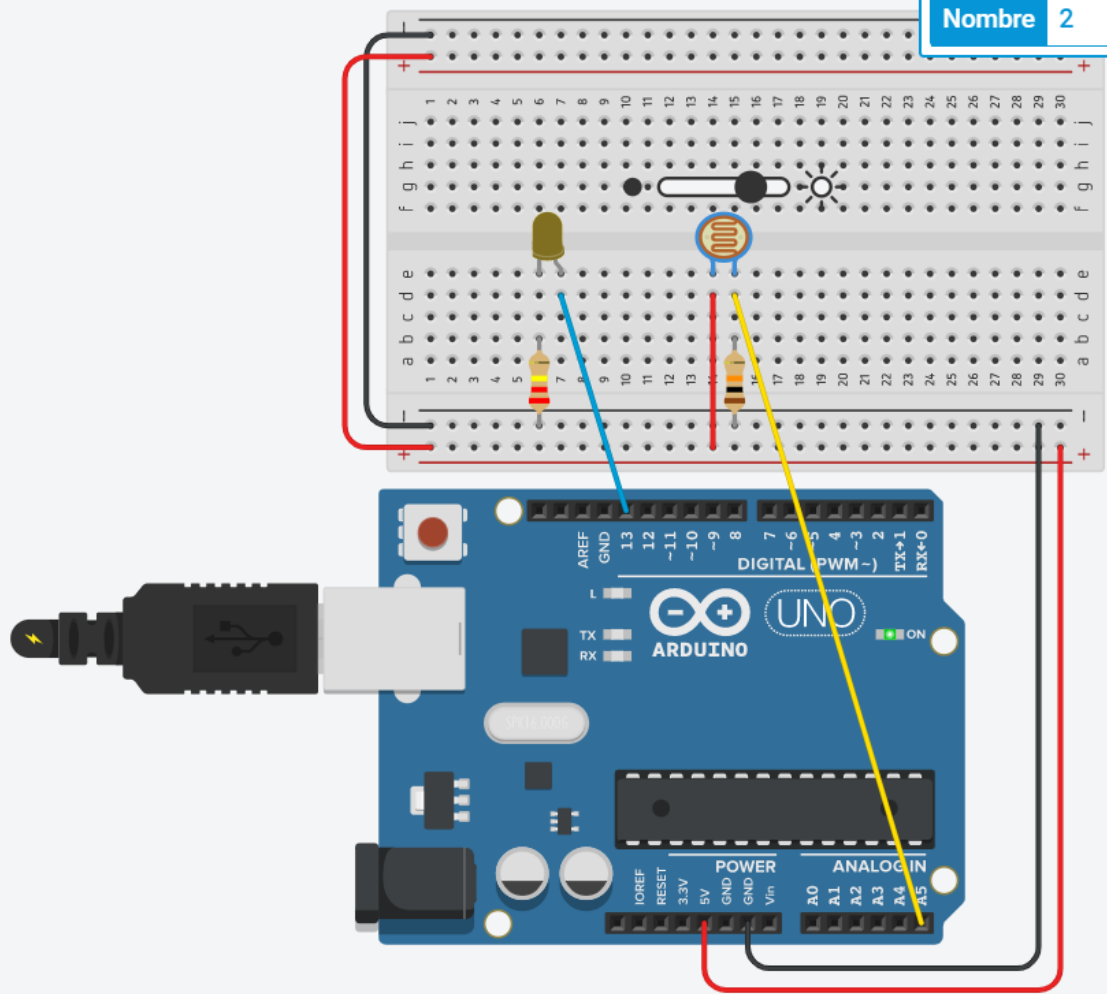
void loop()
{
  if (analogRead(LDR) < 900) digitalWrite(LED, HIGH);
  else digitalWrite(LED, LOW);
  delay(500);
}
```

DEMOSTRACION DEL RESULTADO:



Fotorresistencia

Nombre 2



3. Finalmente, controlar un servomotor con un potenciómetro el grado de giro.
CODIGO:

stunning_gaaris1 Arduino 1.8.13 (Windows Store 1.8.42.0)

Archivo Editar Programa Herramientas Ayuda

```

stunning_gaaris1
#include <Servo.h>
Servo helice;
int grados;

void setup()
{
  Serial.begin(9600);
  helice.attach(2);
}

void loop()
{
  grados = map(analogRead(A5), 0 ,1023,0 , 180);
  Serial.println(grados);
  delay(500);
  helice.write(grados);
}

```

DEMOSTRACION DEL RESULTADO:

TINKER
KIDS
CAD

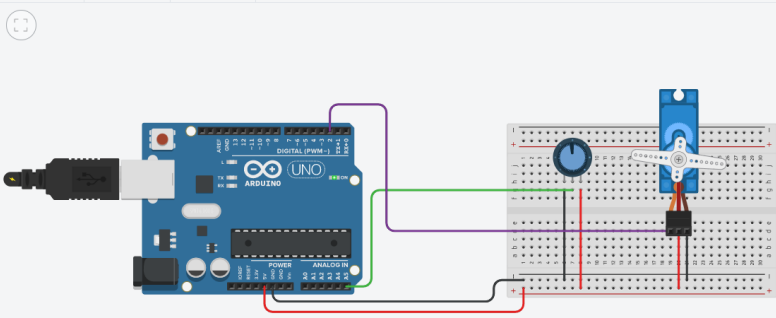
Stunning Gaaris

Se han guardado todos los cambios.

Hora de simulador: 00:00:21

Código Detener simulación Exportar Compartir

1 (Arduino Uno R3)



```

1 #include <Servo.h>
2 Servo helice;
3 int grados;
4
5
6 void setup()
7 {
8   Serial.begin(9600);
9   helice.attach(2);
10 }
11
12 void loop()
13 {
14   grados = map(analogRead(A5), 0 ,1023,0 , 180);
15   Serial.println(grados);
16   delay(500);
17   helice.write(grados);
18 }

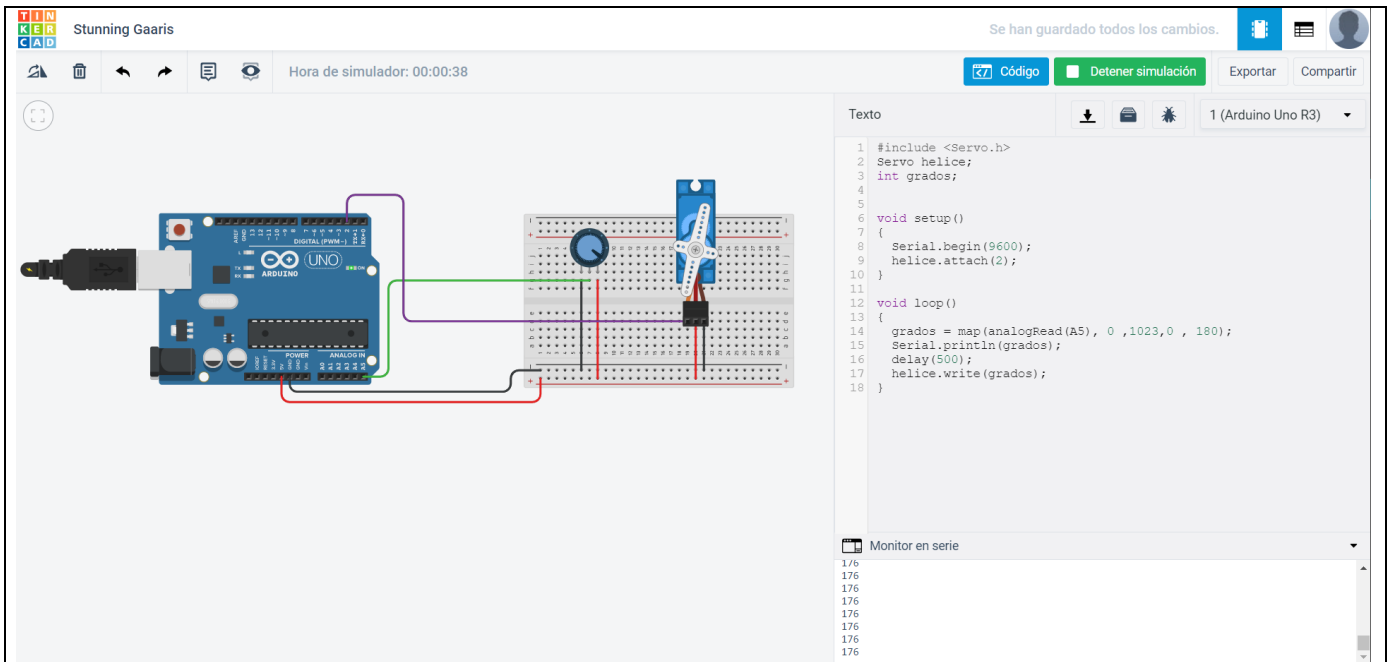
```

Monitor en serie

86
86
86
86
86
86
86

Env. Borrar

<https://www.tinkercad.com/dashboard?type=circuits&collection=designs>



RESULTADO(S) OBTENIDO(S):

- Interpreta de forma correcta la programación en Arduino.
- Identifica correctamente qué herramientas de electrónica se pueden aplicar.

CONCLUSIONES:

- Los estudiantes implementan soluciones de hardware en sistemas.
- Los estudiantes están en la capacidad de implementar sistemas electrónicos en Arduino.

RECOMENDACIONES:

- Revisar la información proporcionada por el docente previo a la práctica.
- Haber asistido a las sesiones de clase.
- **Consultar con el docente las dudas que puedan surgir al momento de realizar la práctica.**

Nombre de estudiante:

PAUL ALEXANDER GUAPUCAL CARDENAS

Firma de estudiante:

