



ugr | Universidad
de **Granada**

PDIH

PERIFÉRICOS Y DISPOSITIVOS DE INTERFAZ HUMANA.

Seminario: Introducción al uso de Arduino.

Autora: Cristina María Crespo Arco

Correo: cmcrespo@correo.ugr.es

Profesor: Pedro A. Castillo Valdivieso



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

Curso 2021 - 2022

Índice

1. Primer ejercicio.	2
1.1. Descripción del ejercicio	2
1.2. Componentes eléctricos.	2
1.3. Código.	2
1.4. Esquema de conexiones eléctricas.	3
1.4.1. Vista de Circuito.	3
1.4.2. Vista esquemática.	3
1.5. Funcionamiento del proyecto: En el simulador Tinkercad.	4
1.6. Funcionamiento del proyecto.	4
2. Segundo ejercicio: Versión interruptor.	5
2.1. Descripción del ejercicio	5
2.2. Componentes eléctricos.	5
2.3. Código.	5
2.4. Esquema de conexiones eléctricas.	6
2.4.1. Vista de Circuito.	6
2.4.2. Vista esquemática.	6
2.5. Funcionamiento del proyecto: En el simulador Tinkercad.	7
3. Segundo ejercicio: Versión pulsador.	8
3.1. Descripción del ejercicio	8
3.2. Componentes eléctricos.	8
3.3. Código.	8
3.4. Esquema de conexiones eléctricas.	9
3.4.1. Vista de Circuito.	9
3.4.2. Vista esquemática.	9
3.5. Funcionamiento del proyecto: En el simulador Tinkercad.	10
3.6. Funcionamiento del proyecto.	11

1. Primer ejercicio.

1.1. Descripción del ejercicio

Implementar el programa de parpadeo de LED, ampliándolo para que encienda y apague alternativamente dos LED (uno rojo y otro verde), conectados a las salidas digitales 12 y 13 del Arduino, a un intervalo de 1.5 segundos. Crear el esquema con Fritzing y cargar el programa en Arduino para comprobar que funciona correctamente.

1.2. Componentes eléctricos.

- 1 Arduino Uno R3
- 2 Resistencias 220 Ω
- 1 LED Verde
- 1 LED Rojo
- 5 cables macho-macho

1.3. Código.

```
// C++ code
void setup(){
    // put your setup code here, to run once:
    pinMode(12, OUTPUT);      //pin de salida de la LED roja
    pinMode(13, OUTPUT);      //pin de salida de la LED verde
}

void loop(){
    // put your main code here, to run repeatedly:
    digitalWrite(13, LOW);    //Apagar LED verde
    digitalWrite(12, HIGH);   //Encender LED amarilla
    delay(1500);             // Esperar durante 1,5 segundos
    digitalWrite(12, LOW);    //Apagar LED amarilla
    digitalWrite(13, HIGH);   //Encender LED verde
    delay(1500);             // Esperar durante 1,5 segundos
}
```

1.4. Esquema de conexiones eléctricas.

1.4.1. Vista de Circuito.

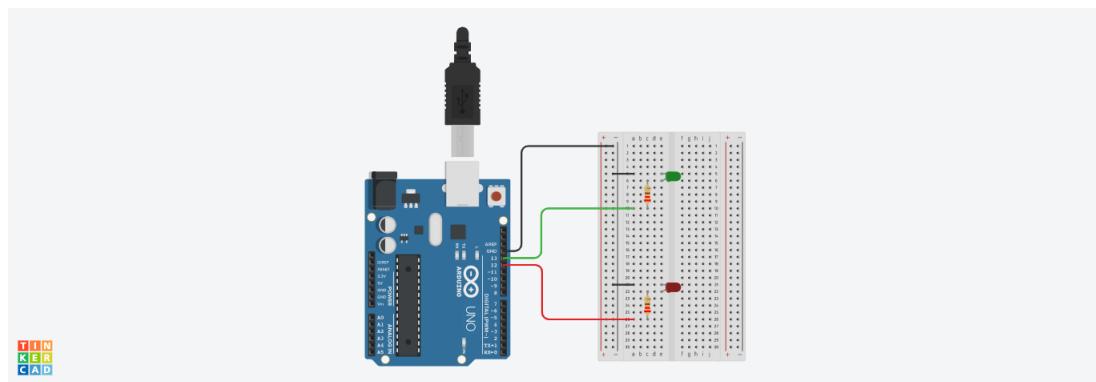


Figura 1: Circuito.

1.4.2. Vista esquemática.

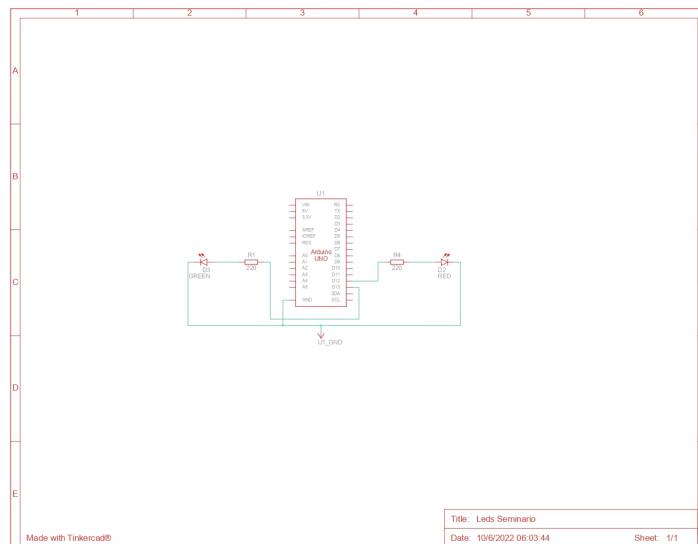


Figura 2: Esquema.

1.5. Funcionamiento del proyecto: En el simulador Tinkercad.

Enlace al proyecto en Tinkercad:

<https://www.tinkercad.com/things/4D4ASgkwvRN-leds-seminario/editel?sharecode=DVNGxVeWVXMLfTkTyUiDQZuPnQXBhPHNpUX4yVN5RTQ>

Enlace al vídeo con una demostración del funcionamiento del proyecto en el simulador:

https://github.com/cmcrespo00/PDIH/blob/main/S-Arduino/ejercicio1/ejerV1_1.mp4

1.6. Funcionamiento del proyecto.

- Circuito.

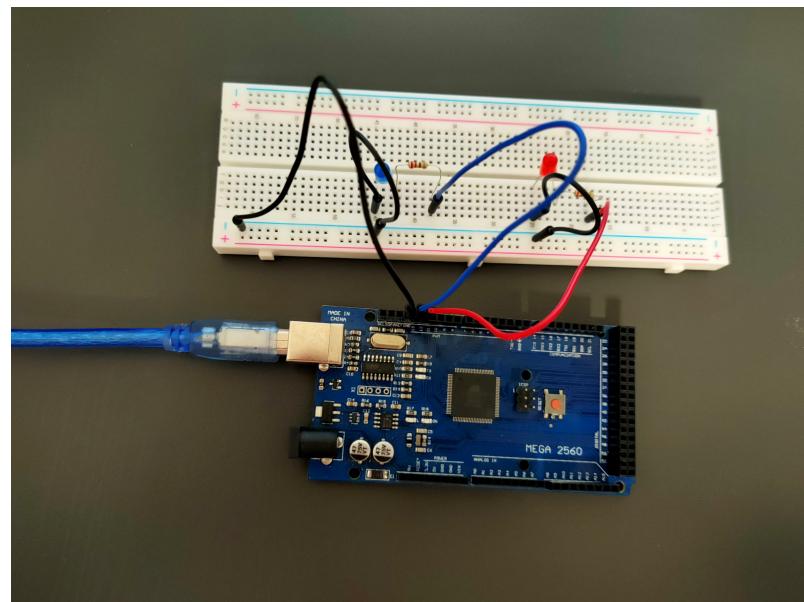


Figura 3: Arduino.

- Enlace al vídeo con una demostración del funcionamiento del proyecto:

https://github.com/cmcrespo00/PDIH/blob/main/S-Arduino/ejercicio1/ejerV1_2.mp4

2. Segundo ejercicio: Versión interruptor.

2.1. Descripción del ejercicio

Implementar en los simuladores el programa de parpadeo de LED, ampliándolo con las modificaciones necesarias para que se encienda el LED solo cuando se pulse un interruptor conectado a la entrada digital 7.

2.2. Componentes eléctricos.

- 1 Arduino Uno R3
- 2 Resistencias 220 Ω
- 1 LED Rojo
- 1 Interruptor deslizante
- 7 cables macho-macho

2.3. Código.

```
// C++ code
void setup(){
    // put your setup code here, to run once:
    pinMode(7, INPUT);           //pin de entrada del INTERRUPTOR
    pinMode(13, OUTPUT);         //pin de salida de la LED roja
}

void loop(){
    // put your main code here, to run repeatedly:
    //Comprueba si el interruptor está encendido
    if ((digitalRead(7) == true)) {
        digitalWrite(13, HIGH);    //Encender LED roja
    }
    //Comprueba si el interruptor está apagado
    else if ((digitalRead(7) == false)) {
        digitalWrite(13, LOW);     //Apagar LED roja
    }
}
```

2.4. Esquema de conexiones eléctricas.

2.4.1. Vista de Circuito.

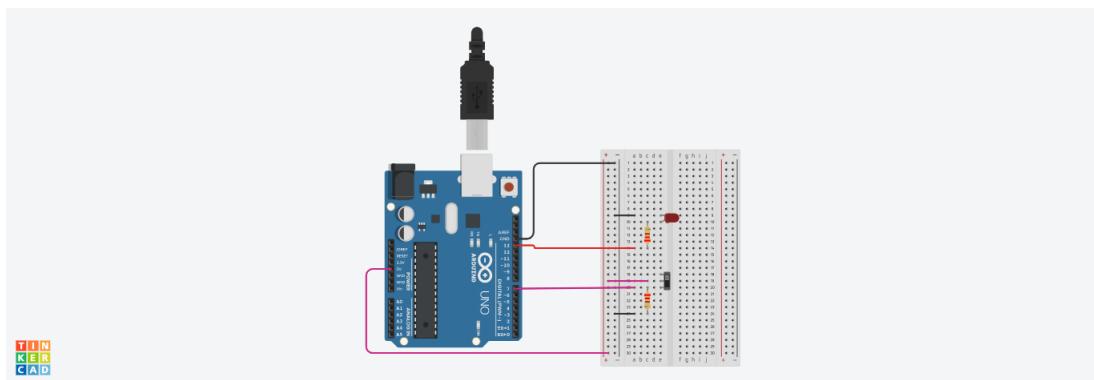


Figura 4: Circuito.

2.4.2. Vista esquemática.

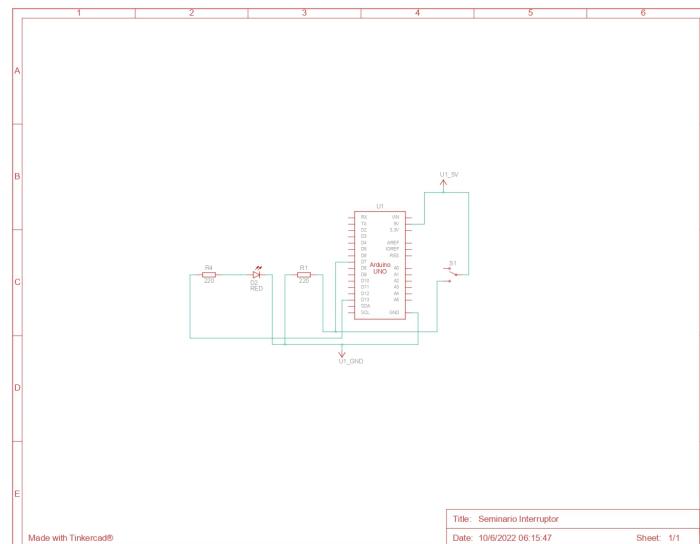


Figura 5: Esquema.

2.5. Funcionamiento del proyecto: En el simulador Tinkercad.

Enlace al proyecto en Tinkercad:

<https://www.tinkercad.com/things/48k5qjj2vhQ-seminario-interruptor/editel?sharecode=4oTdYp3-uYRYC1LEuq-6roYV93r0NAjD868kTYvbCwA>

- Circuito.

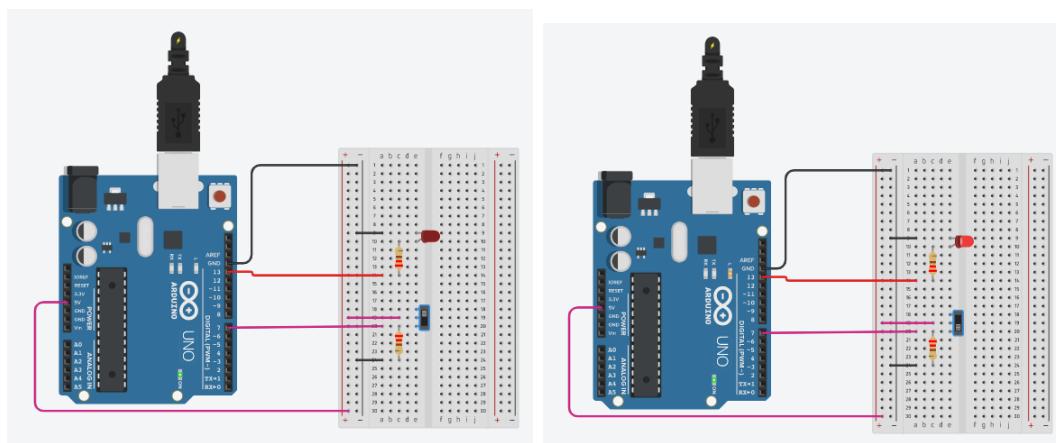


Figura 6: Interruptor apagado - Interruptor encendido.

- Enlace al vídeo con una demostración del funcionamiento del proyecto en el simulador:
https://github.com/cmcrespo00/PDIH/blob/main/S-Arduino/ejercicio2/ejerV2a_1.mp4

3. Segundo ejercicio: Versión pulsador.

3.1. Descripción del ejercicio

Implementar en los simuladores el programa de parpadeo de LED, ampliándolo con las modificaciones necesarias para que se encienda el LED solo cuando se pulse un pulsador conectado a la entrada digital 7.

3.2. Componentes eléctricos.

- 1 Arduino Uno R3
- 2 Resistencias 220 Ω
- 1 LED Rojo
- 1 Pulsador
- 7 cables macho-macho

3.3. Código.

```
// C++ code

void setup(){
    // put your setup code here, to run once:
    pinMode(7, INPUT);           //pin de entrada del PULSADOR
    pinMode(13, OUTPUT);         //pin de salida de la LED roja
}

void loop(){
    // put your main code here, to run repeatedly:
    //Comprueba si el pulsador está pulsado
    if ((digitalRead(7) == true)) {
        digitalWrite(13, HIGH);    //Encender LED roja
    }

    //Comprueba si el pulsador no está pulsado
    else if ((digitalRead(7) == false)) {
        digitalWrite(13, LOW);     //Apagar LED roja
    }
}
```

3.4. Esquema de conexiones eléctricas.

3.4.1. Vista de Circuito.

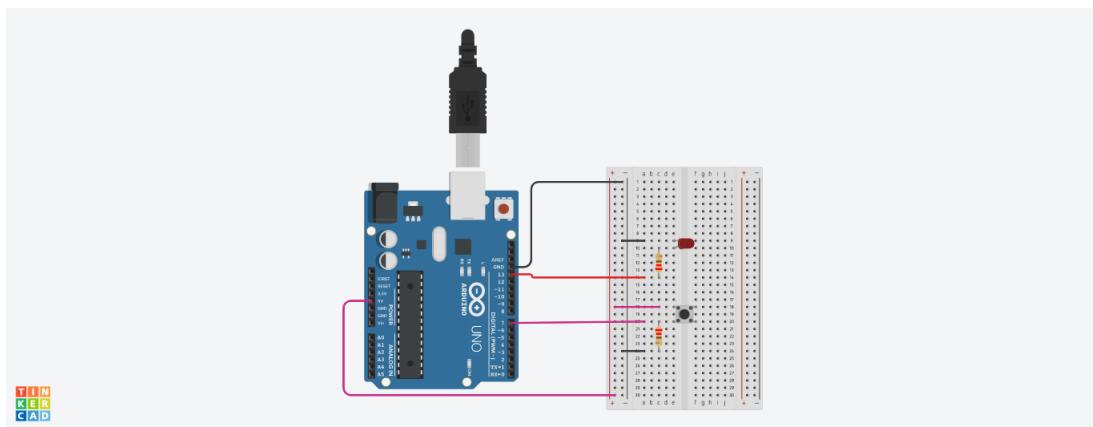


Figura 7: Circuito.

3.4.2. Vista esquemática.

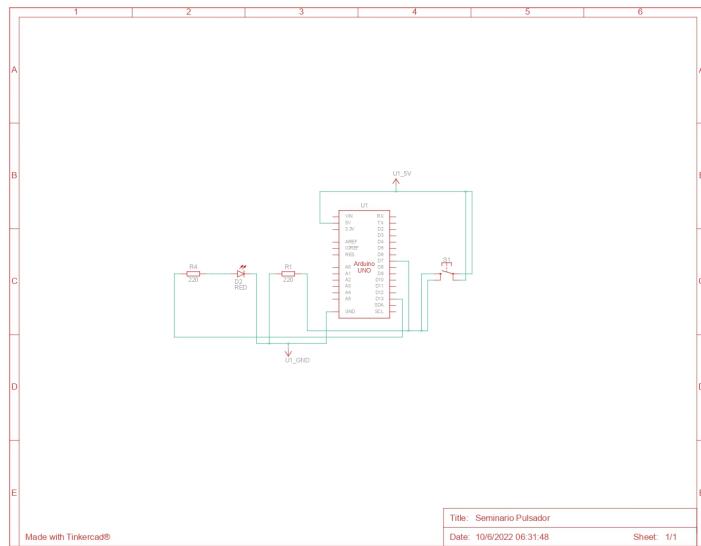


Figura 8: Esquema.

3.5. Funcionamiento del proyecto: En el simulador Tinkercad.

Enlace al proyecto en Tinkercad:

https://www.tinkercad.com/things/2YtDosRtALp-seminario-pulsador/editel?sharecode=WRm8ypvVBrkcqdR0yBaoA8GN_RCoNuGflvTHzR6KggY

- Circuito.

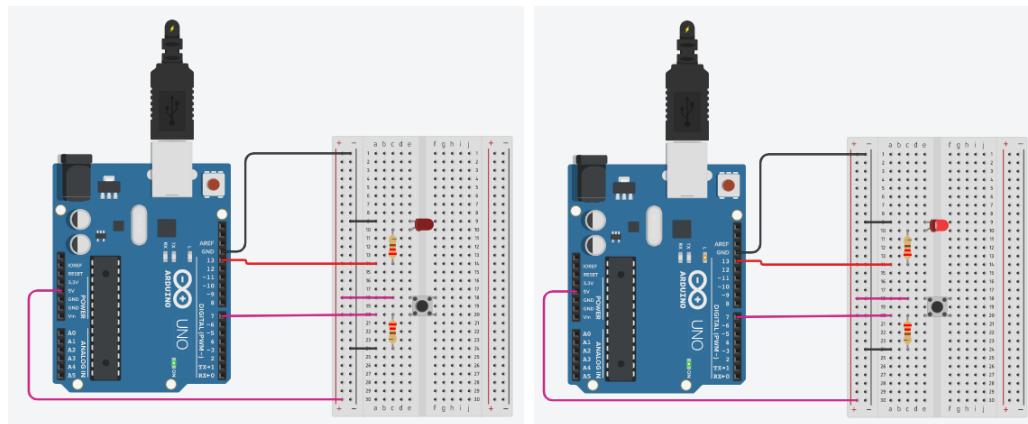


Figura 9: Interruptor apagado - Interruptor encendido.

- Enlace al vídeo con una demostración del funcionamiento del proyecto en el simulador:

https://github.com/cmcrespo00/PDIH/blob/main/S-Arduino/ejercicio2/ejerV2b_1.mp4

3.6. Funcionamiento del proyecto.

- Circuito.

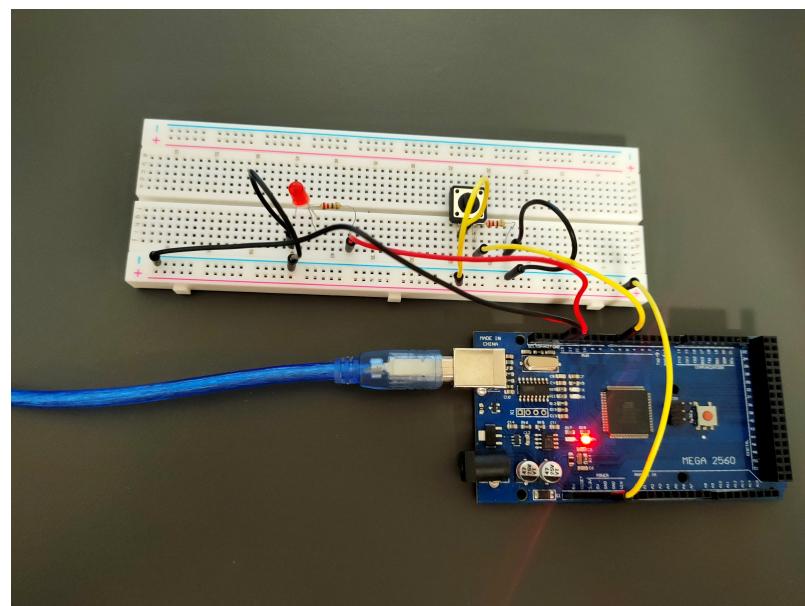


Figura 10: Arduino.

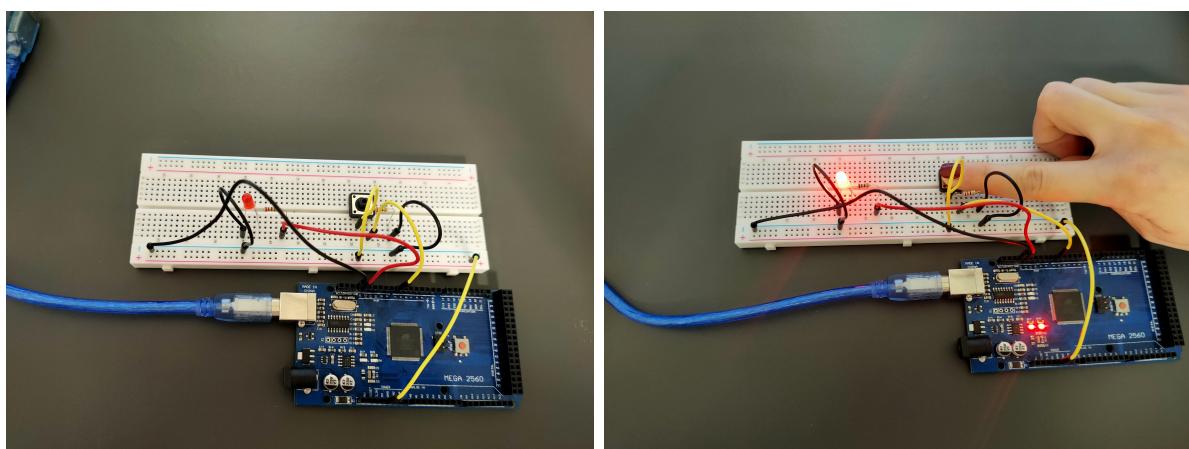


Figura 11: Interruptor apagado - Interruptor encendido.

- Enlace al vídeo con una demostración del funcionamiento del proyecto:

https://github.com/cmcrespo00/PDIH/blob/main/S-Arduino/ejercicio2/ejerV2b_2.mp4