



ugr | Universidad
de **Granada**

PDIH

PERIFÉRICOS Y DISPOSITIVOS DE INTERFAZ HUMANA.

Práctica 3: Experimentación con Arduino.

Autora: Cristina María Crespo Arco

Correo: cmcrespo@correo.ugr.es

Autor: Andrés Piqueras Brück

Correo: andrespquieras@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. Código.	2
1.2.1. Vista de Circuito.	2
1.3. Vista esquemática.	3
1.4. Circuito Arduino.	3
2. Segundo ejercicio.	4
2.1. Descripción del ejercicio	4
2.2. Código.	4
2.2.1. Vista de Circuito.	4
2.3. Vista esquemática.	5
3. Tercer ejercicio.	6
3.1. Descripción del ejercicio	6
3.2. Código.	6
3.2.1. Vista de Circuito.	7
3.3. Vista esquemática.	7
3.4. Circuito Arduino.	8
4. Cuarto ejercicio.	9
4.1. Descripción del ejercicio	9
4.2. Código.	9
4.2.1. Vista de Circuito.	9
4.3. Vista esquemática.	10
4.4. Circuito Arduino.	10

1. Primer ejercicio.

1.1. Descripción del ejercicio

Implementar el programa de parpadeo de LED, ampliándolo para que encienda y apague alternativamente tres LEDs (uno rojo, otro amarillo y otro verde), conectados a las salidas digitales 11, 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. Código.

```
// C++ code
void setup(){
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);
}

void loop(){
    digitalWrite(13, LOW);
    (11, HIGH);
    delay(1500); // Wait for 1,5 second(s)
    digitalWrite(11, LOW);
    digitalWrite(12, HIGH);
    delay(1500); // Wait for 1,5 second(s)
    digitalWrite(12, LOW);
    digitalWrite(13, HIGH);
    delay(1500); // Wait for 1,5 second(s)
}
```

1.2.1. Vista de Circuito.

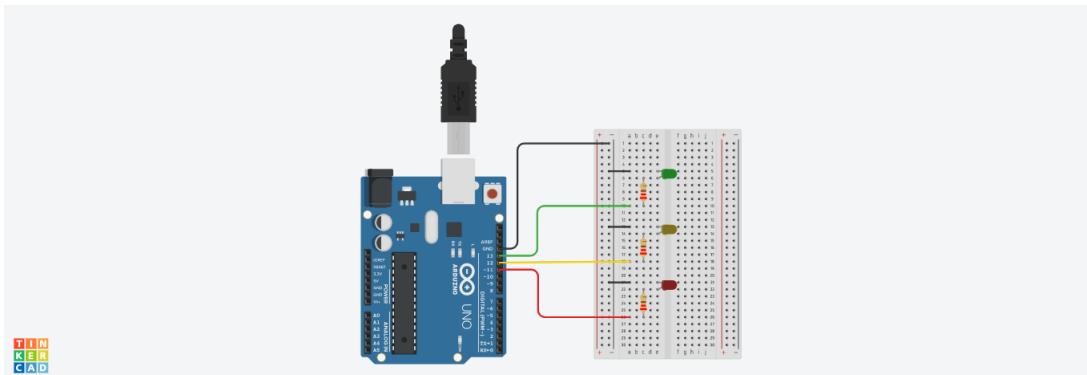


Figura 1: Circuito.

1.3. Vista esquemática.

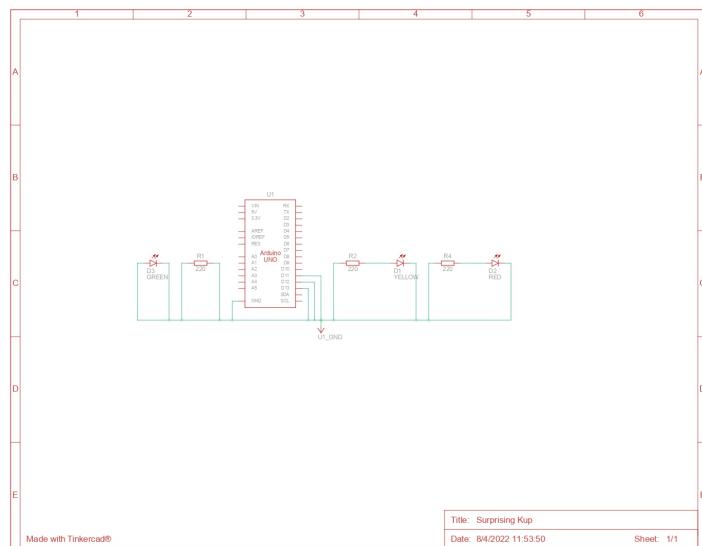


Figura 2: Esquema.

1.4. Circuito Arduino.

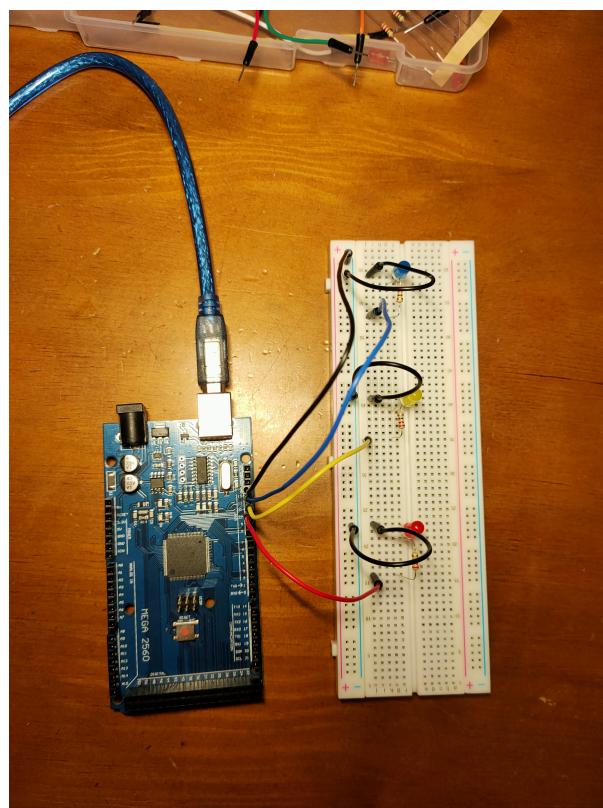


Figura 3: Arduino.

2. Segundo ejercicio.

2.1. Descripción del ejercicio

Partir del programa de parpadeo de LEDs anterior y ampliarlo con las modificaciones necesarias para que se encienda el LED rojo solo cuando se pulse un interruptor conectado a la entrada digital 7, y en ese momento se apaguen los LEDs amarillo y verde.

2.2. Código.

```
// C++ code
void setup(){
    pinMode(7, INPUT);
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);
}

void loop(){
    if ((digitalRead(7) == true)) {
        digitalWrite(11, HIGH);
        digitalWrite(12, LOW);
        digitalWrite(13, LOW);
    } else if ((digitalRead(7) == false)) {
        digitalWrite(11, LOW);
        digitalWrite(12, HIGH);
        digitalWrite(13, HIGH);
    }
}
```

2.2.1. Vista de Circuito.

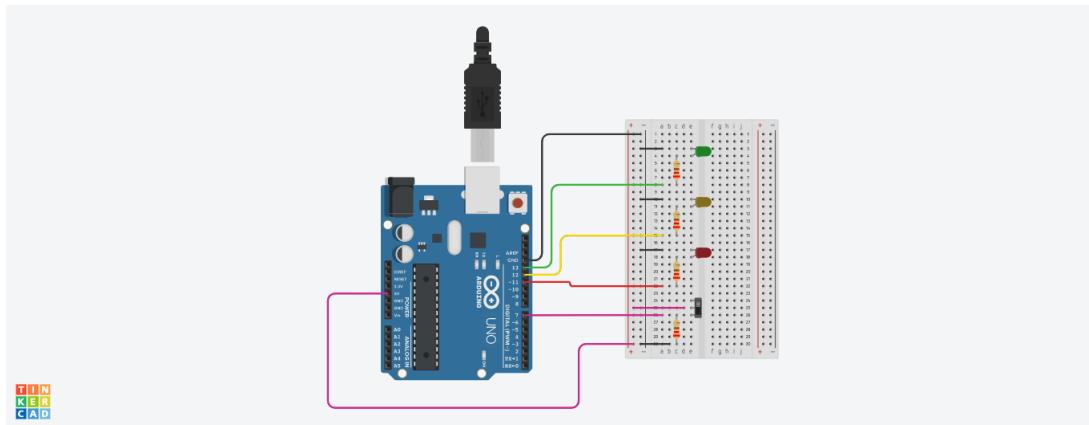


Figura 4: Circuito.

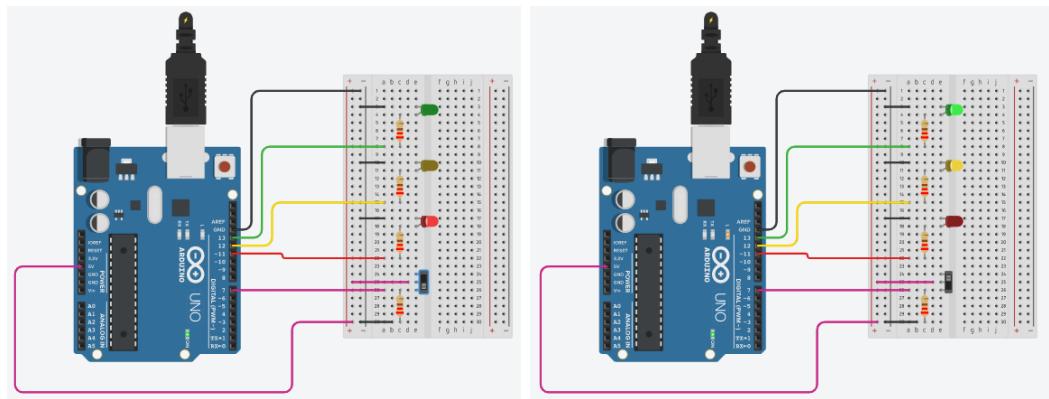


Figura 5: Interruptor encendido - Interruptor apagado.

2.3. Vista esquemática.

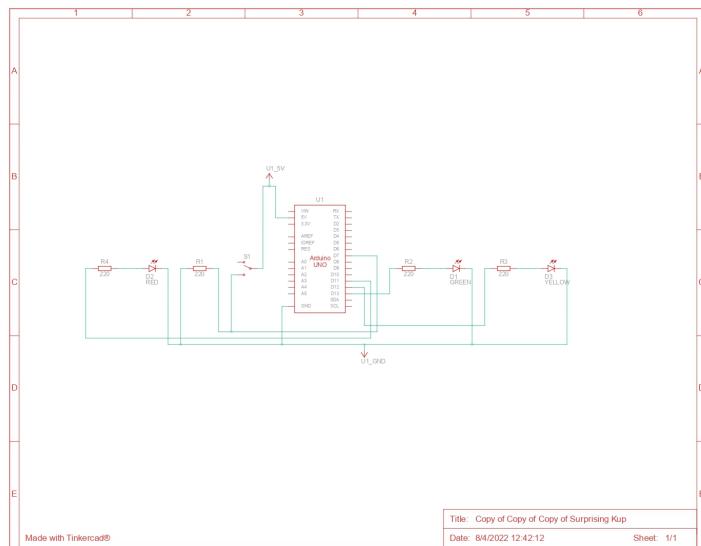


Figura 6: Esquema.

3. Tercer ejercicio.

3.1. Descripción del ejercicio

Secuencia de LEDs, encendiendo y apagando 4 LEDs secuencialmente, de forma similar a las lucecitas de “El coche fantástico”.

3.2. Código.

```
// C++ code
void setup(){
    pinMode(10, OUTPUT);
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);
}

void loop(){
    digitalWrite(11, LOW);
    digitalWrite(10, HIGH);
    delay(200); // Wait for 0,2 second(s)
    digitalWrite(10, LOW);
    digitalWrite(11, HIGH);
    delay(200); // Wait for 0,2 second(s)
    digitalWrite(11, LOW);
    digitalWrite(12, HIGH);
    delay(200); // Wait for 0,2 second(s)
    digitalWrite(12, LOW);
    digitalWrite(13, HIGH);
    delay(200); // Wait for 0,2 second(s)
    digitalWrite(13, LOW);
    digitalWrite(12, HIGH);
    delay(200); // Wait for 0,2 second(s)
    digitalWrite(12, LOW);
    digitalWrite(11, HIGH);
    delay(200); // Wait for 0,2 second(s)
}
```

3.2.1. Vista de Circuito.

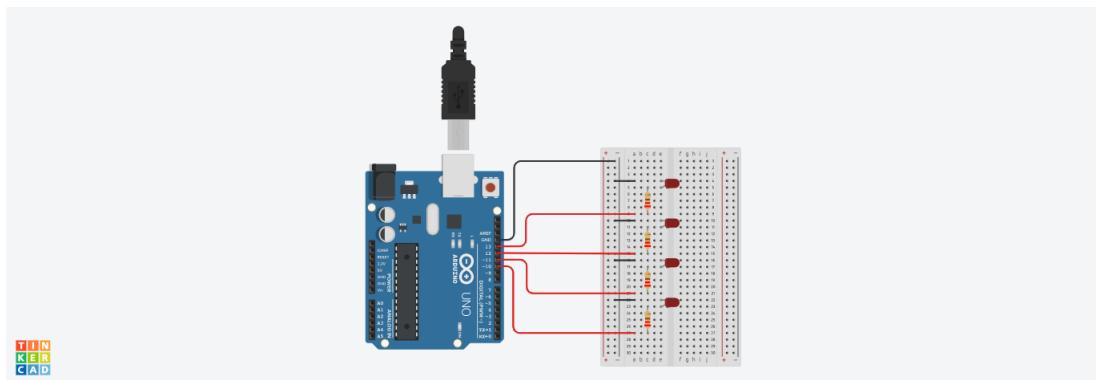


Figura 7: Circuito.

3.3. Vista esquemática.

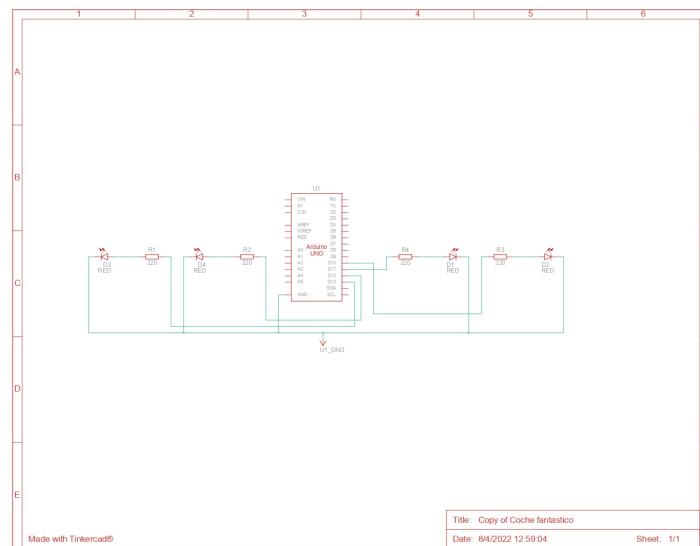


Figura 8: Esquema.

3.4. Circuito Arduino.

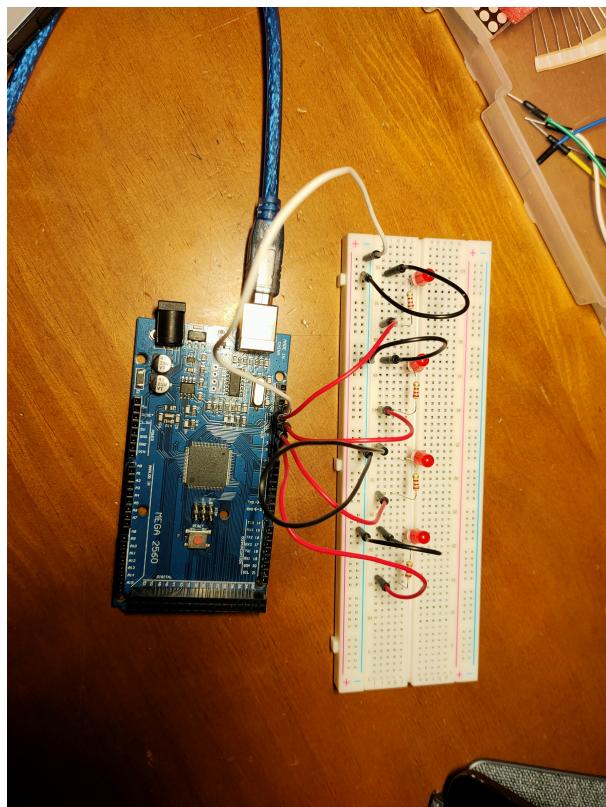


Figura 9: Arduino.

4. Cuarto ejercicio.

4.1. Descripción del ejercicio

Alarma por detección de presencia.

4.2. Código.

```
// C++ code
void setup(){
    pinMode(13, OUTPUT);
    pinMode(7, INPUT);
}

void loop(){
    if ((digitalRead(7) == HIGH)) {
        digitalWrite(13, HIGH);
    } else {
        digitalWrite(13, LOW);
    }
}
```

4.2.1. Vista de Circuito.

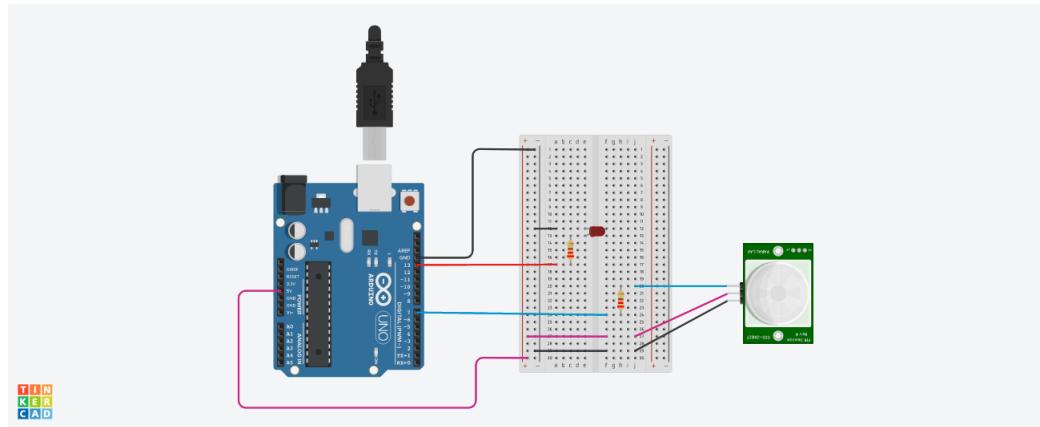


Figura 10: Circuito.

4.3. Vista esquemática.

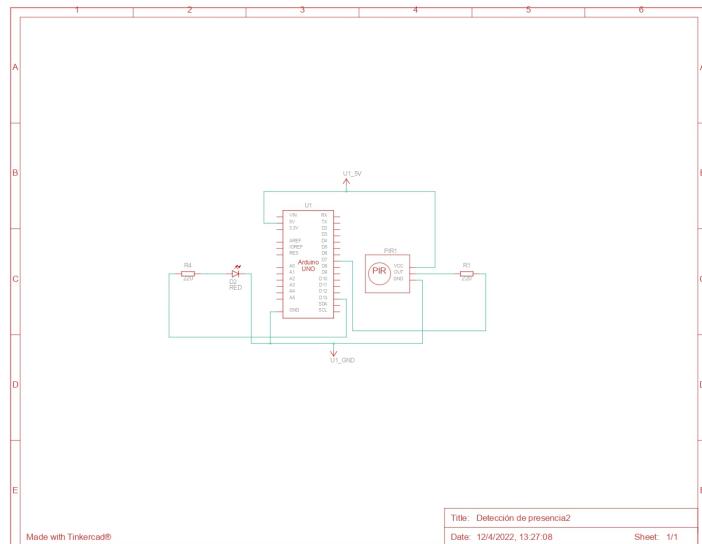


Figura 11: Esquema.

4.4. Circuito Arduino.

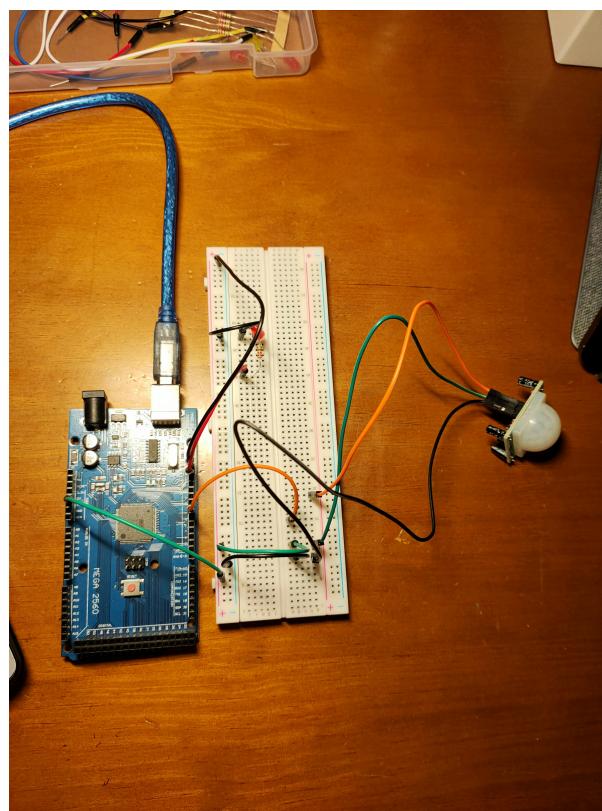


Figura 12: Arduino.