

Práctica 5. Implementación Parpadeo de un LED con Arduino UNO

Esta es la primera practica de implementación, la cual consiste en hacer parpadear un LED por medio de Arduino UNO.

Objetivo

Hacer parpadear un LED utilizando Arduino UNO.

Equipo y Materiales

1 Arduino UNO

1 Protoboard

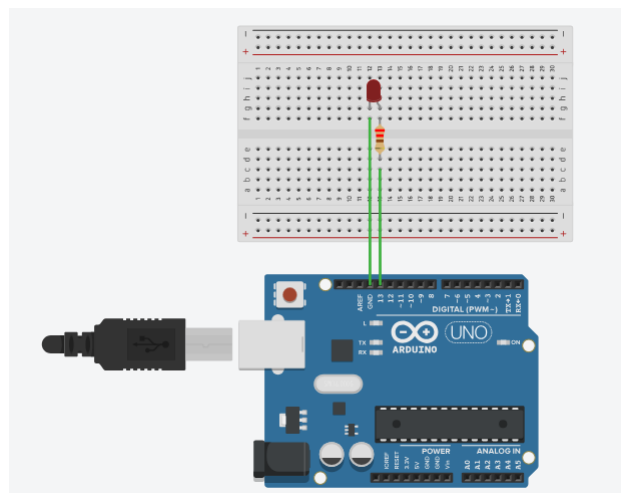
1 Resistencia de 220Ω

1 LED

IDE Arduino

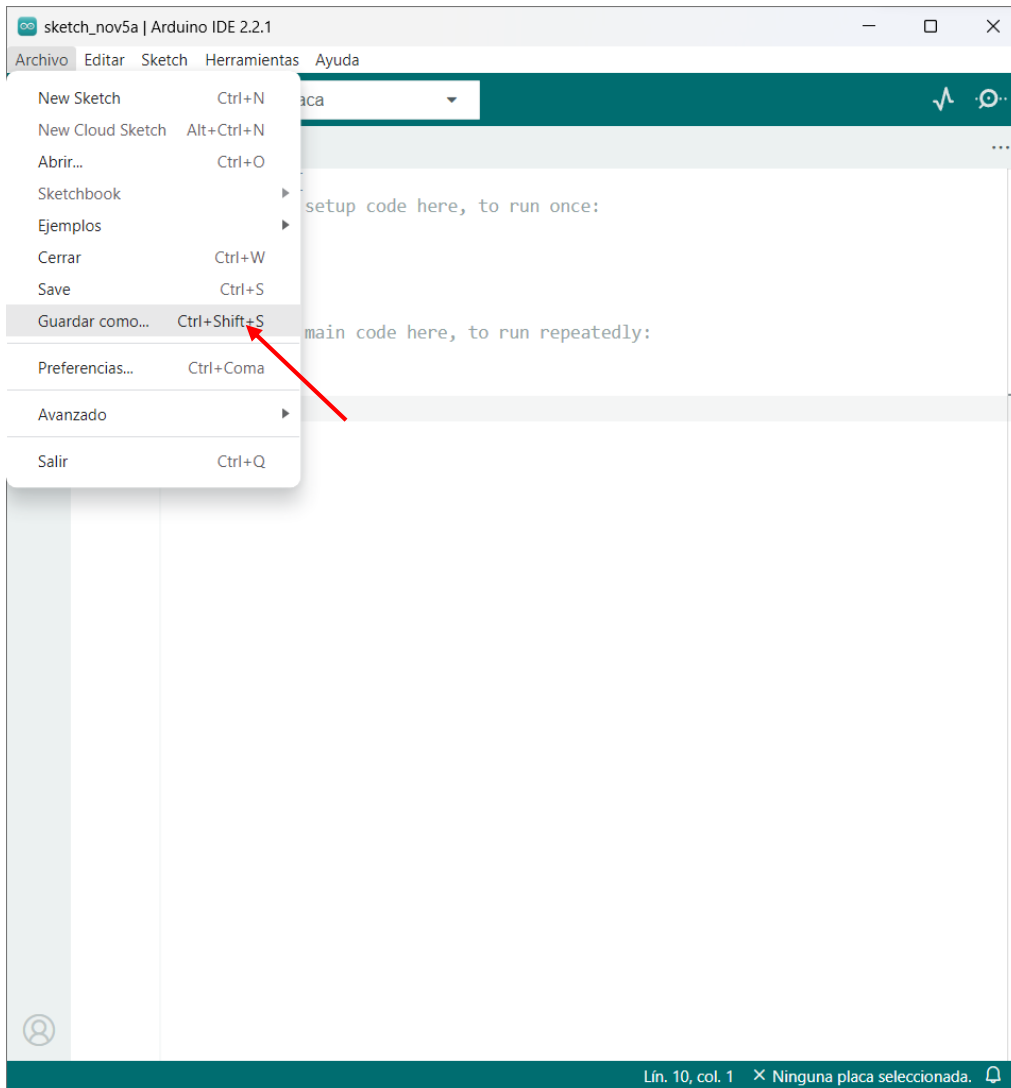
Procedimiento

PASO 1. Realice el circuito con sus componentes físicos como se muestra en el ejemplo:



Implementación

PASO 2. Diríjase al IDE Arduino. Siempre al iniciar deberá de nombrar su proyecto. En la parte superior de click sobre “Archivo” y en “guardar como”, borre el nombre predefinido y sustitúyalo por “blinkNoDelay.ino”



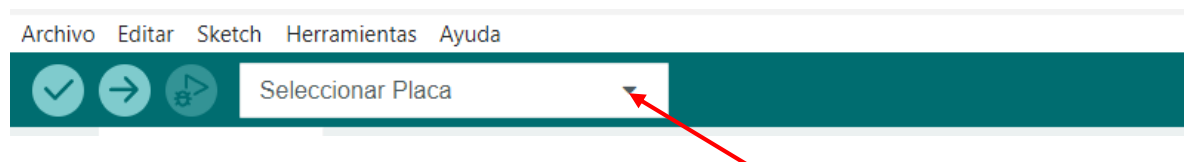
PASO 3. Una vez guardado el archivo, borre el contenido que le aparece y sustitúyalo por el siguiente código.

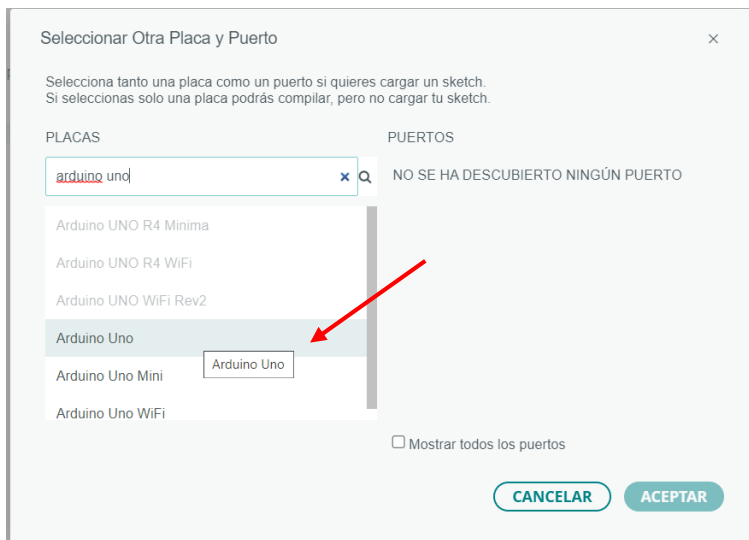
```

1  /*
2  * blinkNoDelay.ino
3  *
4  * Este programa hace que el led de status del Arduino UNO o un le
5  * conectado al pin 13, parpadee, sin usar la funcion delay().
6  * Esto permite que otro codigo ejecute al mismo tiempo sin ser
7  * interrumpido por el codigo del LED. Los periodos de estar
8  * encendido y apagado del led son iguales
9  */
10 const unsigned int PIN_LED = 13;
11 // Periodo en ms que dura encendido o apagado el LED
12 const long PERIODO = 1000;
13 int estadoLed = LOW;
14 // Almacena la hora en la que se prendio/apago el LED
15 unsigned long lapsoAnterior = 0;
16
17 void setup() {
18     // Establece el pin PIN_LED como de salida:
19     pinMode(PIN_LED, OUTPUT);
20 }
21
22 void loop() {
23     // Obtiene la hora actual
24     unsigned long lapsoActual = millis();
25     // Verifica si es tiempo de prender/apagar el LED
26     if (lapsoActual - lapsoAnterior >= PERIODO) {
27         //Almacena la hora en la que se prendio/apago el LED
28         lapsoAnterior = lapsoActual;
29         if (estadoLed == LOW) {
30             estadoLed = HIGH;
31         }
32         else {
33             estadoLed = LOW;
34         }
35     }
36     else {
37         digitalWrite(PIN_LED, estadoLed);
38     }
39 }

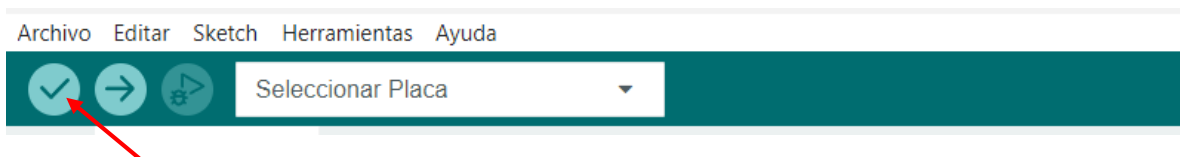
```

PASO 4. Seleccione la placa que está utilizando



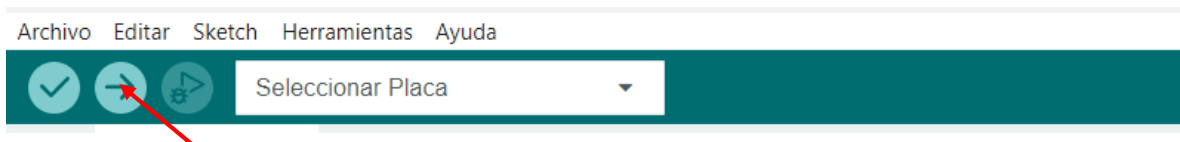


PASO 5. Verifique que su programa no tenga errores.



PASO 6. Conecte el cable del Arduino a este y al puerto USB de su computadora.

PASO 7. Cargue el programa



CUESTIONARIO

1. ¿Qué método puede utilizar para identificar la polaridad del LED?
2. ¿Cuál es la diferencia entre utilizar `delay()` y el método que utilizamos en esta práctica con los condicionales?