



Módulo 2: Ejercicios Arduino

Sistema Empotrados, Distribuidos y Ubícuos

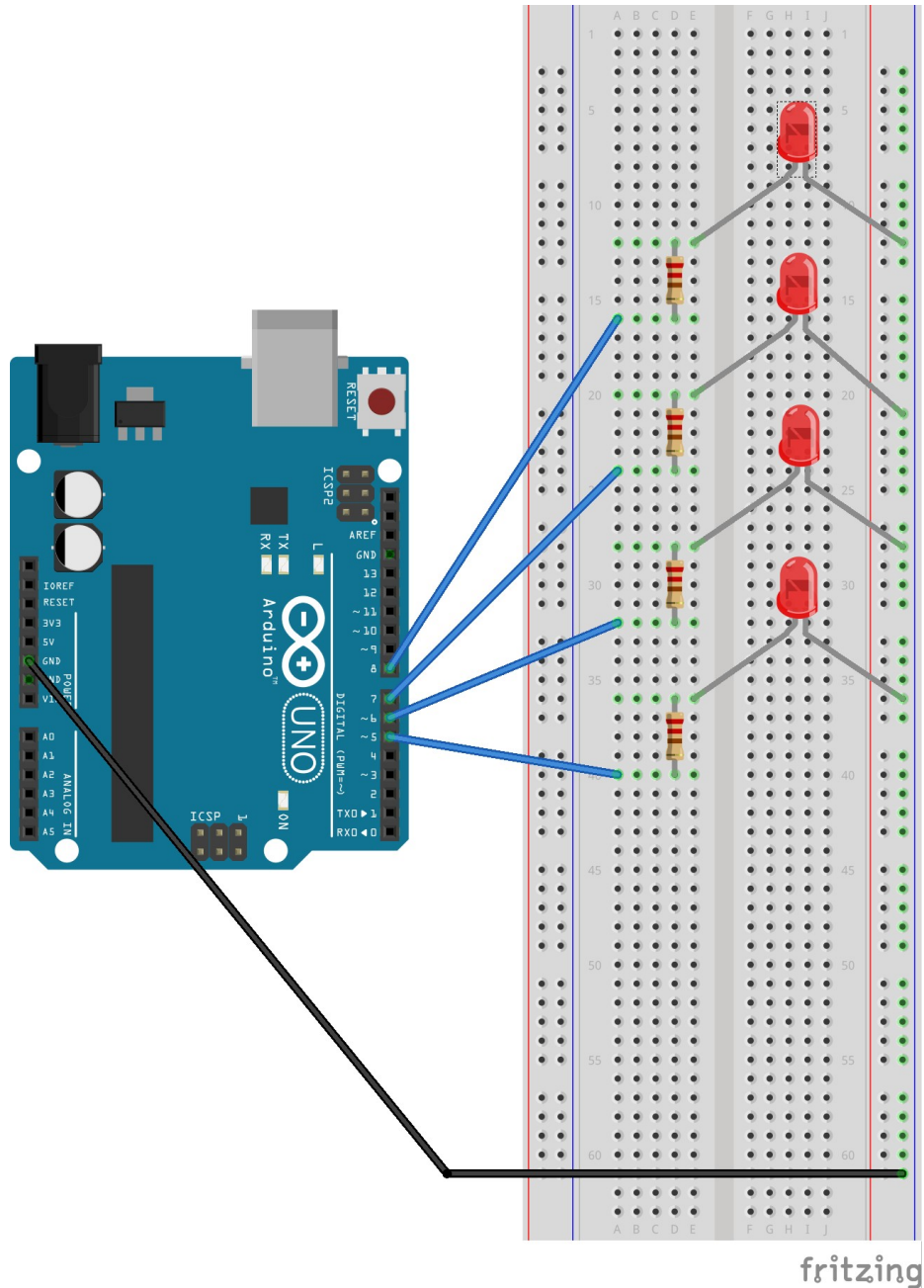
RAFAEL ESCUDERO LIRIO

Secuencial (escudero_ej1)

En este programa se define una función conteniendo dos bucles que encienden y apagan los 4 LEDs del circuito de manera secuencial durante 200ms.

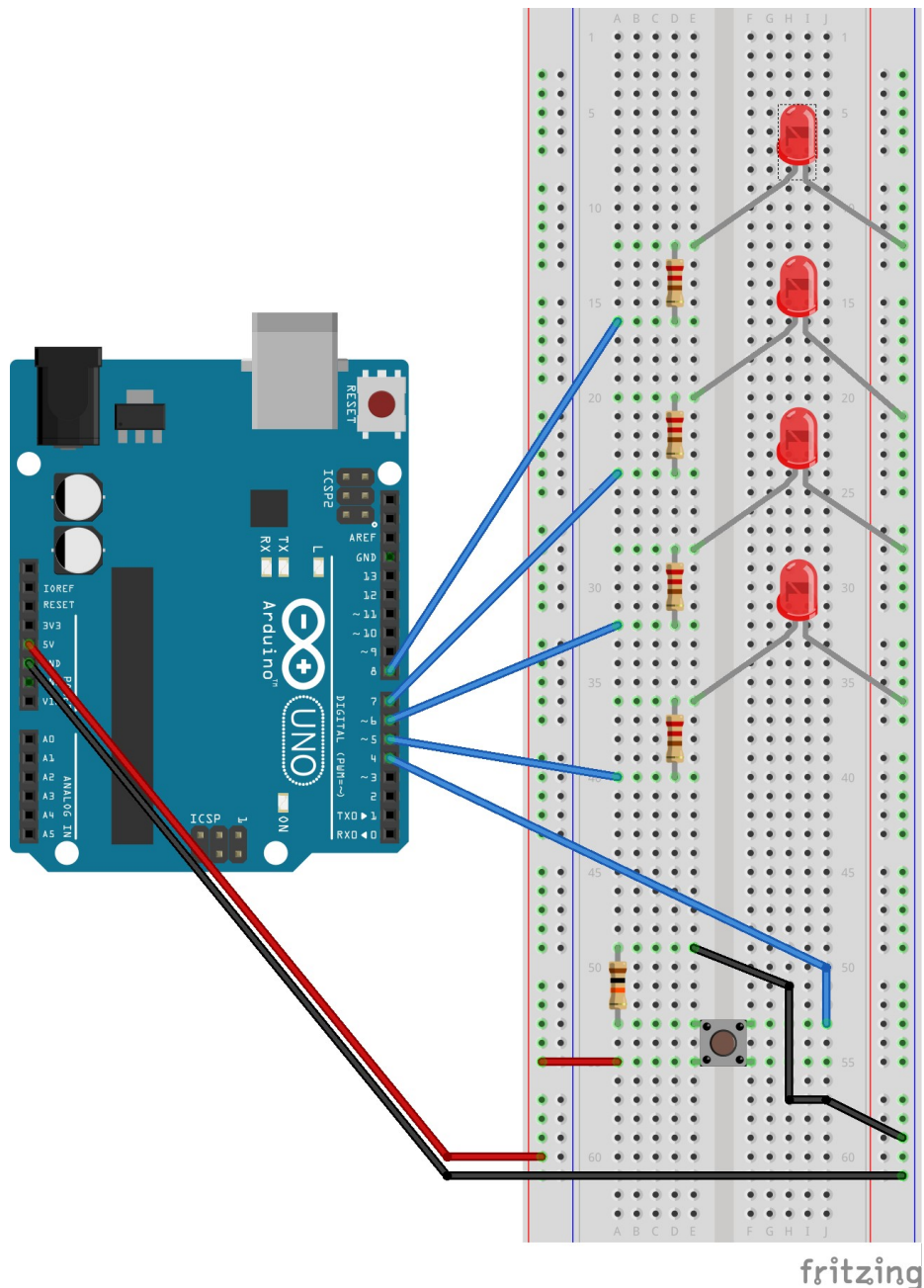
Como puede observarse, al tamaño del array en los bucles debemos dividirlo por el tamaño de una variable int o de otro tipo dado el caso.

Esto es debido a que la función `sizeof()` devuelve el tamaño del array en bytes y un int en arduino ocupa 2 bytes.



SwitchSec (escudero_ej2)

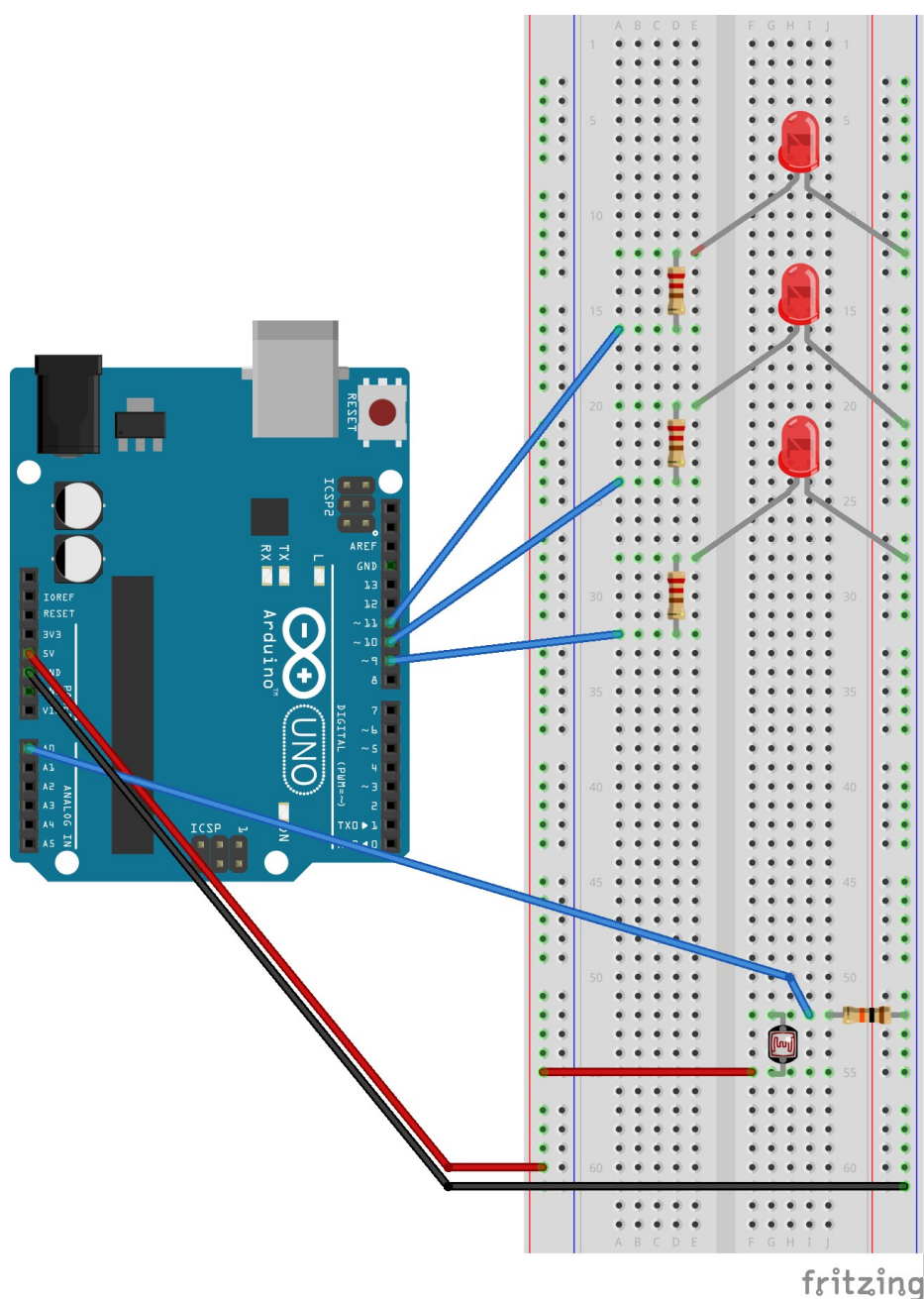
En este programa vamos a añadir la funcionalidad de un switch al secuenciador de LEDs. Haremos una comprobación mediante un bloque if y la secuencia de LEDs del bucle se ejecutará únicamente si el switch está pulsado. Si la secuencia de iluminación aún no ha terminado cuando dejamos de pulsar el switch ésta sigue ejecutándose hasta terminar la iteración actual del bucle.



LDRMeasure (escudero_ej3)

En este programa hacemos uso de una resistencia variable LDR. Este tipo de resistencia es sensible a la luz y nos ofrecerá un valor de lectura analógica diferente según la incidencia de la luz sobre su superficie. Una vez sabemos este valor lo usaremos para determinar la intensidad de brillo de 3 LEDs conectados a puertos digitales con funcionalidad PWM. Gracias a esta funcionalidad asignaremos un valor entre 0 y 255 a cada LED según en rango de valor marcado por la resistencia LDR.

El resultado será una variación dinámica de la luminosidad de todos los LEDs al mismo tiempo. A más oscura quede la habitación donde se encuentra el sensor mayor será la intensidad mostrada por nuestros LEDs.



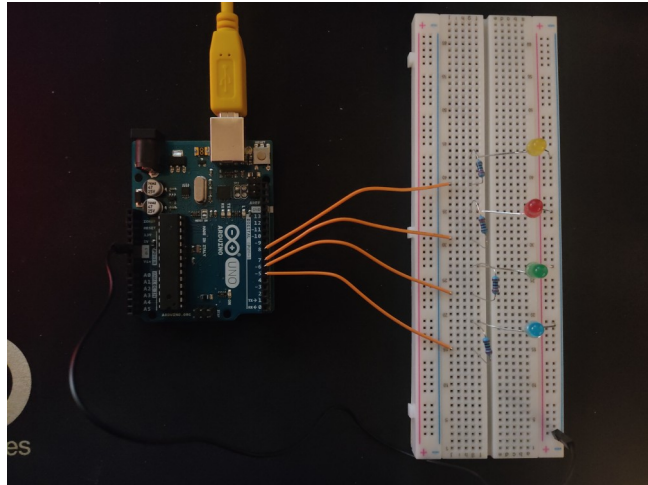


Figura 1: Secuencial (escudero_ej1)

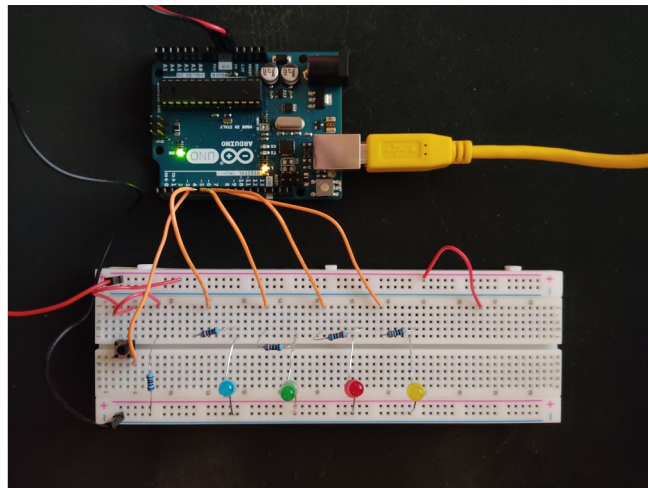


Figura 2: SwitchSec (escudero_ej2)

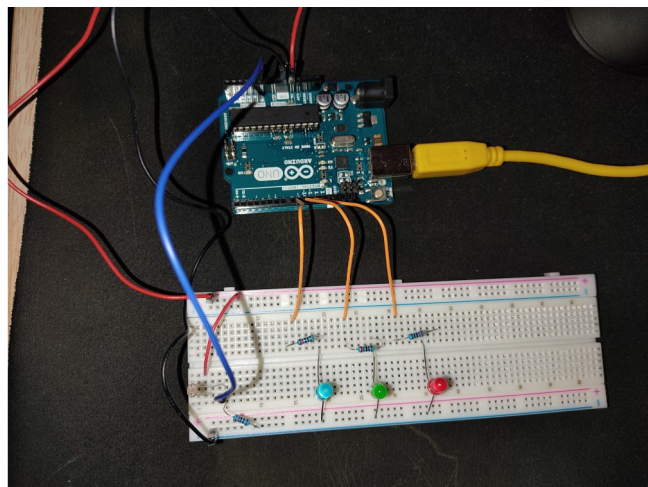


Figura 3: LDRMeasure (escudero_ej3)

ANEXO CON CÓDIGOS

escudero_ej1

```
int leds[4]= {5,6,7,8}; //Definimos el array de int que contiene los identificadores de los leds.
```

```
void setup() {  
    for(int i=0; i<sizeof(leds)/sizeof(int); i++){ //Definimos el modo de los pines como outputs, para ello recorremos el array una vez usando la función pinMode().  
        pinMode(leds[i], OUTPUT);  
    }  
}
```

```
void loop() {  
    secuencial(); //Llamada a nuestro método.  
}
```

```
void secuencial(){  
    for(int i=0; i<sizeof(leds)/sizeof(int); i++){ //En este método recorremos un bucle for en el cual encendemos el led, esperamos 200ms y apagamos ese led.  
        digitalWrite(leds[i], HIGH); //El bucle recorre las 4 posiciones del array con lo que los leds se encenderán y apagarán de forma secuencial.  
        delay(200);  
        digitalWrite(leds[i], LOW);  
    }  
}
```

escudero_ej2

```
int SWITCH = 4; //Definimos el pin del switch

int leds[4]= {5,6,7,8};

void setup() {
    for(int i=0; i<sizeof(leds)/sizeof(int); i++){
        pinMode(leds[i], OUTPUT);
    }
    pinMode(SWITCH, INPUT); //Definimos el modo del pin como input, lo
    usaremos como entrada.
}

void loop() {
    secuencial();
}

void secuencial(){
    if(digitalRead(SWITCH) == 1){ //Haciendo uso del método
    digitalWrite() podemos leer el pin del switch. Al estar pulsado el
    switch se cumple la condición y entra en el bucle.
        for(int i=0; i<sizeof(leds)/sizeof(int); i++){
            digitalWrite(leds[i], HIGH);
            delay(200);
            digitalWrite(leds[i], LOW);
        }
    }
}
```


escudero_ej3

```
int LDR = A0; //Definimos el pin de lectura de la resistencia.
int leds[3]= {9,10,11};
int valLDR = 0; //Inicializamos la variable en la que guardaremos el
valor de lectura.

void setup() {
    Serial.begin(9600); //Inicializamos el puerto serie con un valor de
refresco.
}

void loop() {
    valLDR = analogRead(LDR); //La función analogRead() nos permite
leer valores analógicos.
    Serial.println(valLDR); //Vamos a mostrar el valor analógico de la
resistencia en el puerto serie.
    //Aquí definimos 4 bloques condicionales en los que, tras leer el
valor de la resistencia LDR haremos uso de la función analogWrite()
sobre los pins de nuestros LEDS.
    //Con el fin de reutilizar código vamos a hacer uso del bucle for de
escritura sobre pines.
    for(int i=0; i<sizeof(leds)/sizeof(int); i++){
        if(valLDR>=768 && valLDR<=1023){
            analogWrite(leds[i], 64);
        }
        if(valLDR>=512 && valLDR<=767){
            analogWrite(leds[i], 127);
        }
        if(valLDR>=256 && valLDR<511){
            analogWrite(leds[i], 191);
        }
        if(valLDR>=0 && valLDR<=255){
            analogWrite(leds[i], 255);
        }
    }
}
```