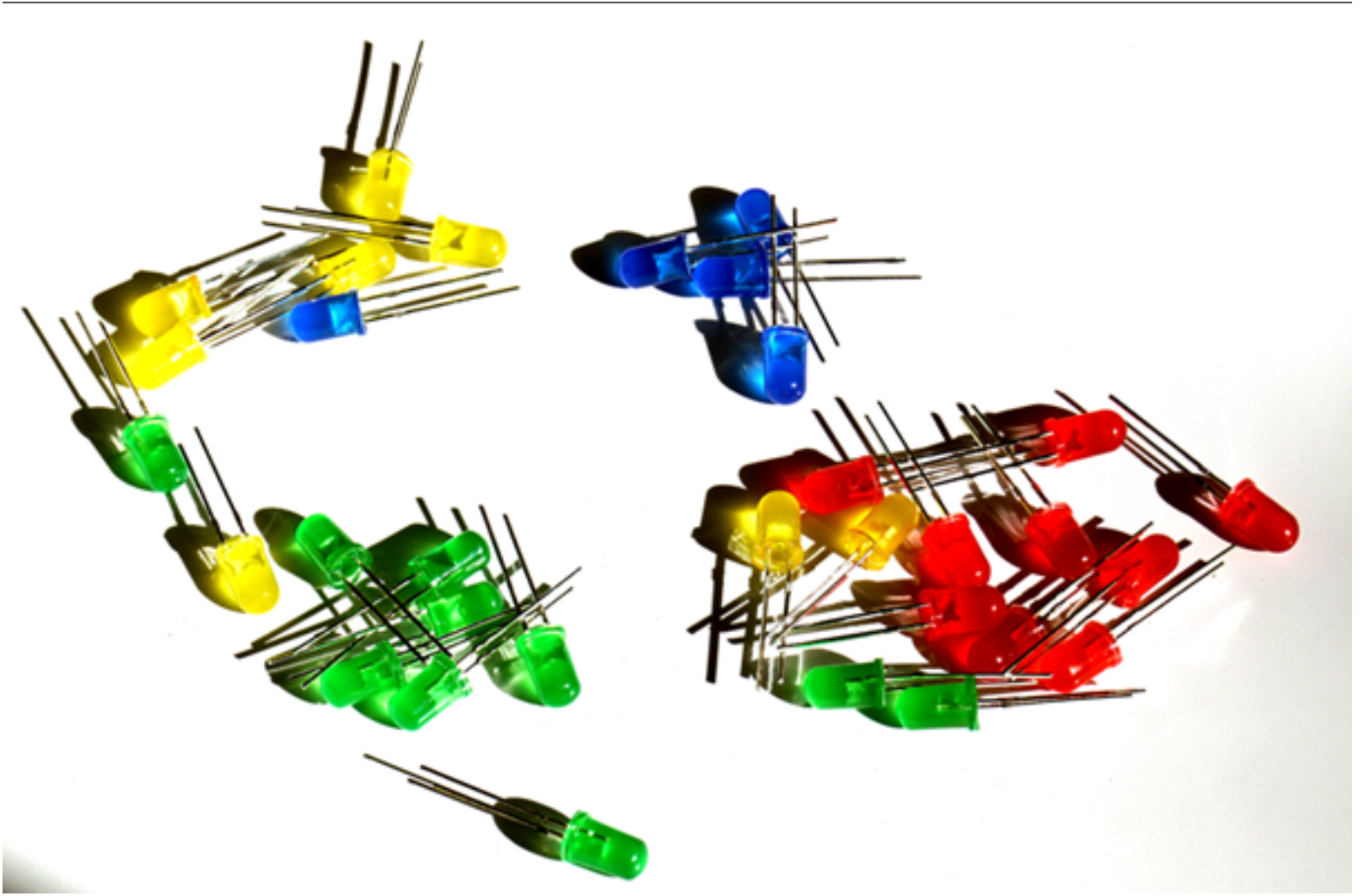
**LED**



**ÍNDICE**

**LED**

Pág

1.- ¿Qué es un Led? 4

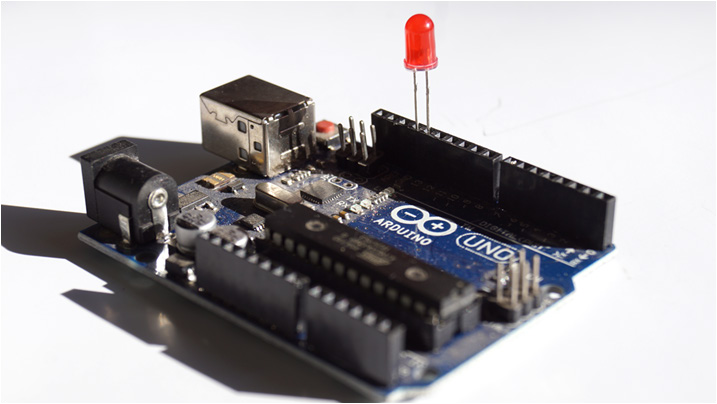
2.- Encender un Led con Arduino 5

3.- Encender una secuencia de Led 8

4.- Encender Led con secuencia For 10

5.- Encender Led con Array 11

6.- Encender Led con PWM (modulación por ancho de pulso) 12



**1.- ¿Qué es un Led ?**

El Led es un diodo emisor de luz (conocido por la sigla LED, (del inglés ***Light- Emitting Diode***). Es una fuente de luz constituida por un material semiconductor dotado de dos terminales.

Un diodo es un componente electrónico que permite la circulación de la corriente eléctrica a través de él en un solo sentido, bloqueando el paso si la corriente circula en sentido contrario, es decir que conduce mejor en un sentido que en el otro; un diodo ideal presenta resistencia nula en un sentido de la corriente y una resistencia infinita en el otro.

Por lo tanto el Led tiene una polaridad, hay que conectarlo en el sentido correcto, su pata más larga correspon- de al Anodo(+) y la mas corta al Catodo(-) de acuerdo a la siguiente figura:

**(+) (-)**

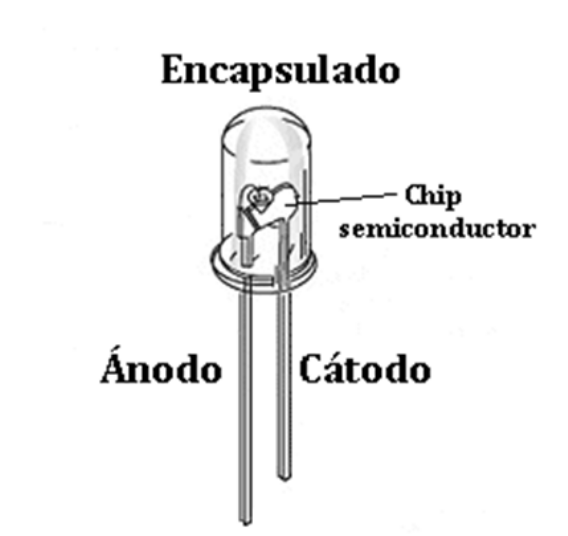


Fig. 1

Un led comienza a emitir luz cuando se le aplica una tensión de 2-3 voltios.

Y los encontramos de diferentes longitudes de onda, o sea de diversos colores

Fig. 2

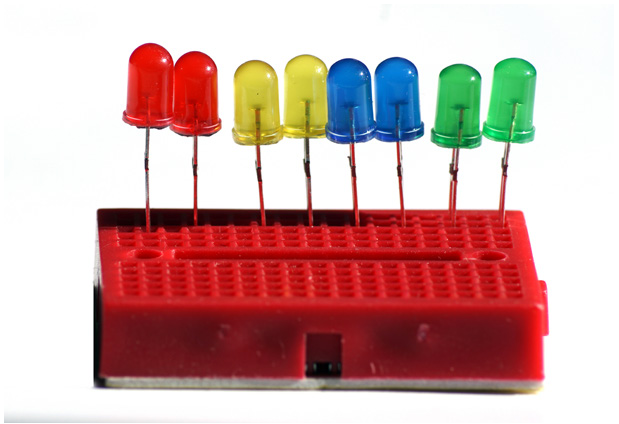


Fig. 2

**2.- Encender un Led con Arduino**

**Objetivo : Encender uno o varios Led, a través de la placa Arduino, con diferentes opciones y tiempos.**

Nuestra primera tarea será la de encender un led a través de nuestra placa Arduino.

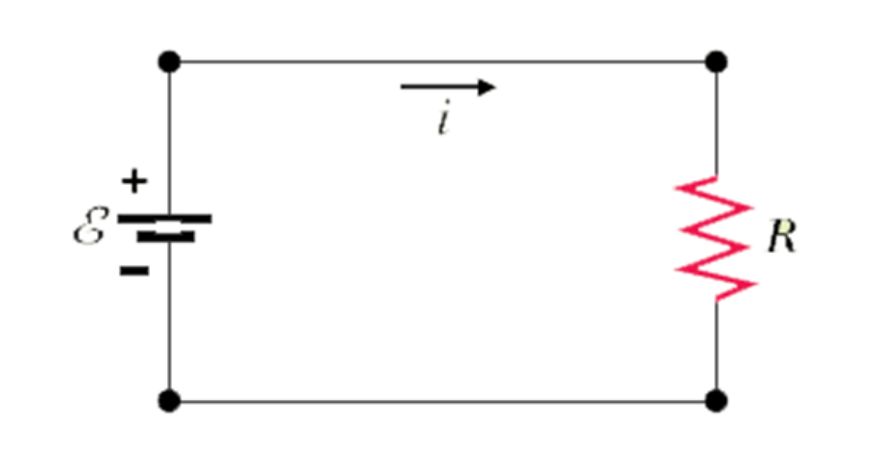
Recordemos que nuestra placa Arduino a través de los pines nos da una salida de 5V CC. Pero los led con esa salida se quemarán, por lo tanto tenemos que considerar una resistencia.

Pero en la placa Arduino tenemos el Pin13, que tiene una resistencia interna, lo cual podemos conectar nues- tro Led de manera directa, en caso contrario se usará una resistencia.

Recordemos algo de la teoría de Circuitos para corrientes continuas.

Recordemos la Ley de Ohm, esta que dice que, la diferencia de potencial es directamente proporcional a la intensidad de la corriente.

Matemáticamente se representa como **V = KI**, donde **K** es una constante, o sea **V = RI**, donde V es la diferen- cia de potencial, **R** es el valor de la esistencia e **I** es la intensidad de corriente.



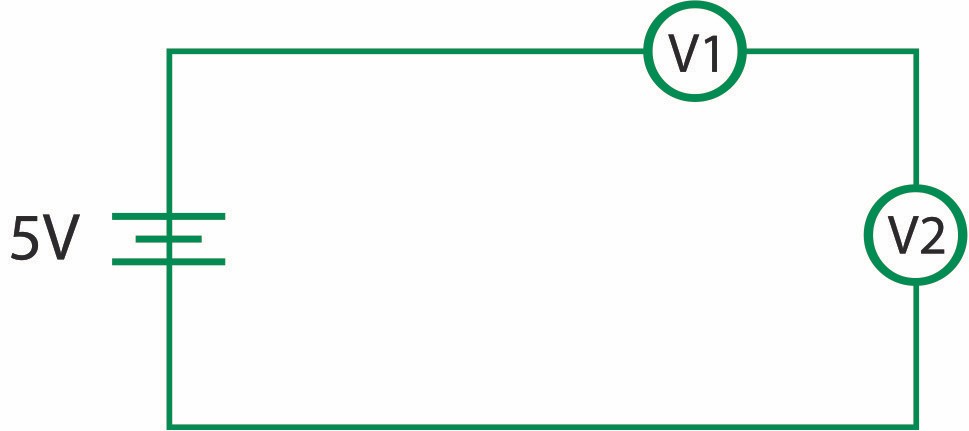
Carasterísticas técnicas de la placa Arduino

Fig. 3



Como nuestra placa Arduino nos entrega una tensión de 5 V y el Led Rojo funciona con una tensión de 1,9 V, al conectarla directamente a la placa éste se nos quemaría, por lo tanto tenemos que considerar agregar una resistencia al circuito para que este Led funcione normalmente.

Consideremos el siguiente circuito:



La suma de las tensiones parciales es igual a la tensión total aplicada al circuito

Vt = V1 + V2 y como Vt = 5

5 = V1 + V2

Como V = R\*I tenemos que: V1 = R1\*I1

V2 = R2\*I2

La intensidad de corriente es la misma, por lo tanto: I1=I2 que corresponde a los 40 mA o 0,04 A

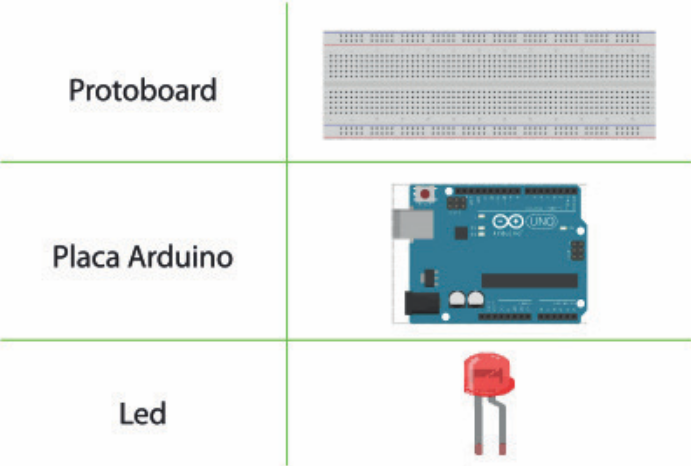
V1 = R1\*I V2 = R2\*I

5 = V1(Led) + V2(resistencia)

5= 1,9 + R2\*0,04 (5-1,9) = R2\*0,04 (5-1,9)/0,04 = R2

R2 = 77,5 Ohm

Consideremos de aquí en adelante para mayor seguridad una resistencia de 220 Ohm



Usaremos el pin 13, ya que éste considera una resistencia interna. En caso de usar otro pin, debemos conside- rar una resistencia de 220 Ohm.

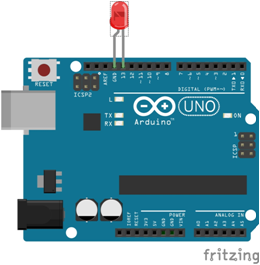


Fig 4

void setup() {

pinMode(13,OUTPUT); // configura el pin 13 como salida

}

void loop() {

digitalWrite(13,HIGH); // envia 5V al pin 13 delay(500);

digitalWrite(13,LOW); // envia 0V al pin 13 delay(500);

}

Qué ocurre si la instrucción delay (500) se cambia por diferentes valores como delay (1000) ó delay (100). Tarea: Realizar la misma experiencia pero variando los delay, considere una variable global.



**3.- Encender una secuencia de Led**

**Objetivo: Encender una secuencia de led**

Encenderemos una secuencia de Led, a través de la placa Arduino, con diferentes tiempos de encendido y apagado.

Se trata de encender 6 led en forma secuencial, los led estarán conectados a los pines 2,3,4,5,6,7.

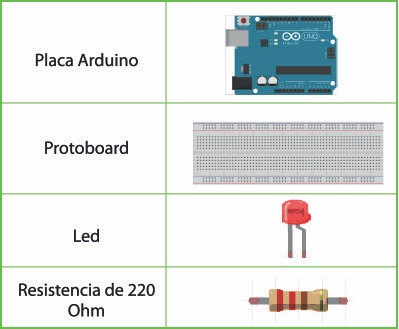
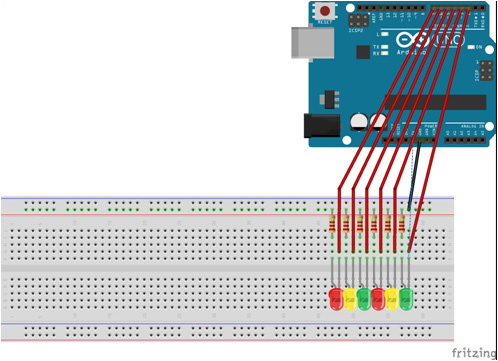


Fig 5

Código



int tiempo=500; // declaramos el tiempo como una variable global, un entero de valor 500 void setup() {

pinMode(2,OUTPUT); // Conectamos a los pines respectivos pinMode(3,OUTPUT);

pinMode(4,OUTPUT); pinMode(5,OUTPUT); pinMode(6,OUTPUT);



pinMode(7,OUTPUT);

}

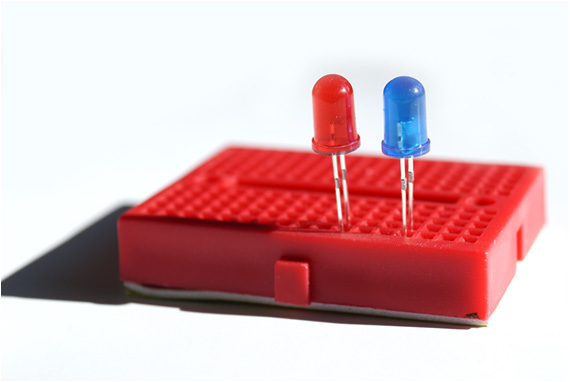
void loop() {

digitalWrite(2,HIGH); // Enciende Led delay(tiempo);

digitalWrite(2,LOW); // Apaga LED

delay(tiempo); digitalWrite(3,HIGH); delay(tiempo); digitalWrite(3,LOW); delay(tiempo); digitalWrite(4,HIGH); delay(tiempo); digitalWrite(4,LOW); delay(tiempo); digitalWrite(5,HIGH); delay(tiempo); digitalWrite(5,LOW); delay(tiempo); digitalWrite(6,HIGH); delay(tiempo); digitalWrite(6,LOW); delay(tiempo); digitalWrite(7,HIGH); delay(tiempo); digitalWrite(7,LOW); delay(tiempo);

}



**4.- Encender un Led con Secuencia For**

**Objetivo: encender varios led mediante la instrucción For.**

Utilizaremos el mismo ejemplo anterior, pero ahora utilizaremos la instrucción For, explicada en clases ante- riores.

Esta función es una iteracción, y funciona hasta que se cumpla una condición. Recordemos un poco la sentencia For

Ejemplo:

for(n=1; n<=10; n++)

{

sentencias;

}

el n = 1 nos indica desde qué número se inicia la iteracción, en este caso en 1. el n<= 10 es la condición. En este caso, n es menor o igual a 10

n++ significa n = n+1;

Esta sentencia se usará para el mismo ejemplo anterior int tiempo=200; // Declaramos una variable global

int n;

void setup() { //comienza la configuración for (n=2;n<8;n++) {

pinMode (n, OUTPUT);

}

}

En la opción loop vamos a poner una función, que se llamará Prende\_Led();

void loop() { Prende\_Led();

}

Ahora construimos la función Prende\_Led();

void Prende\_Led() {

for (n=2;n<8;n++) { // es decir toma los valores desde n=2 hasta n=7 digitalWrite (n, HIGH);

delay (tiempo); digitalWrite (n, LOW); delay (tiempo);

}

}

Como vemos la cantidad de texto se simplifica enormemente al usar la sentencia For. Podemos variar el tiempo solo cambiando la variable global

Ahora cambiaremos la variable global tiempo = 200 por:

tiempo = 100 tiempo = 50; tiempo = 10;



**5.- Encender un Led mediante un Array**

Ahora ocuparemos una secuencia mediante un arreglo matricial (Array).

Un arreglo (vector, array, matriz) es un conjunto de datos o una estructura de datos homogéneos que se en- cuentran ubicados en forma consecutiva en la memoria (sirve para almacenar datos en forma temporal).

Usando la misma estructura de circuito que el ejemplo anterior (Fig 5) probamos este ejercicios, con la instruc- ción for pero usando un array (arreglo)

int leds[]={2,3,4,5,6,7};// Declara variables tipo lista de valores int tiempo=200;

int n=0;

void setup() { //comienza la configuración for (n=0;n<8;n++) {

pinMode (leds[n], OUTPUT);

}

}

void loop() { Prende\_Led();

}

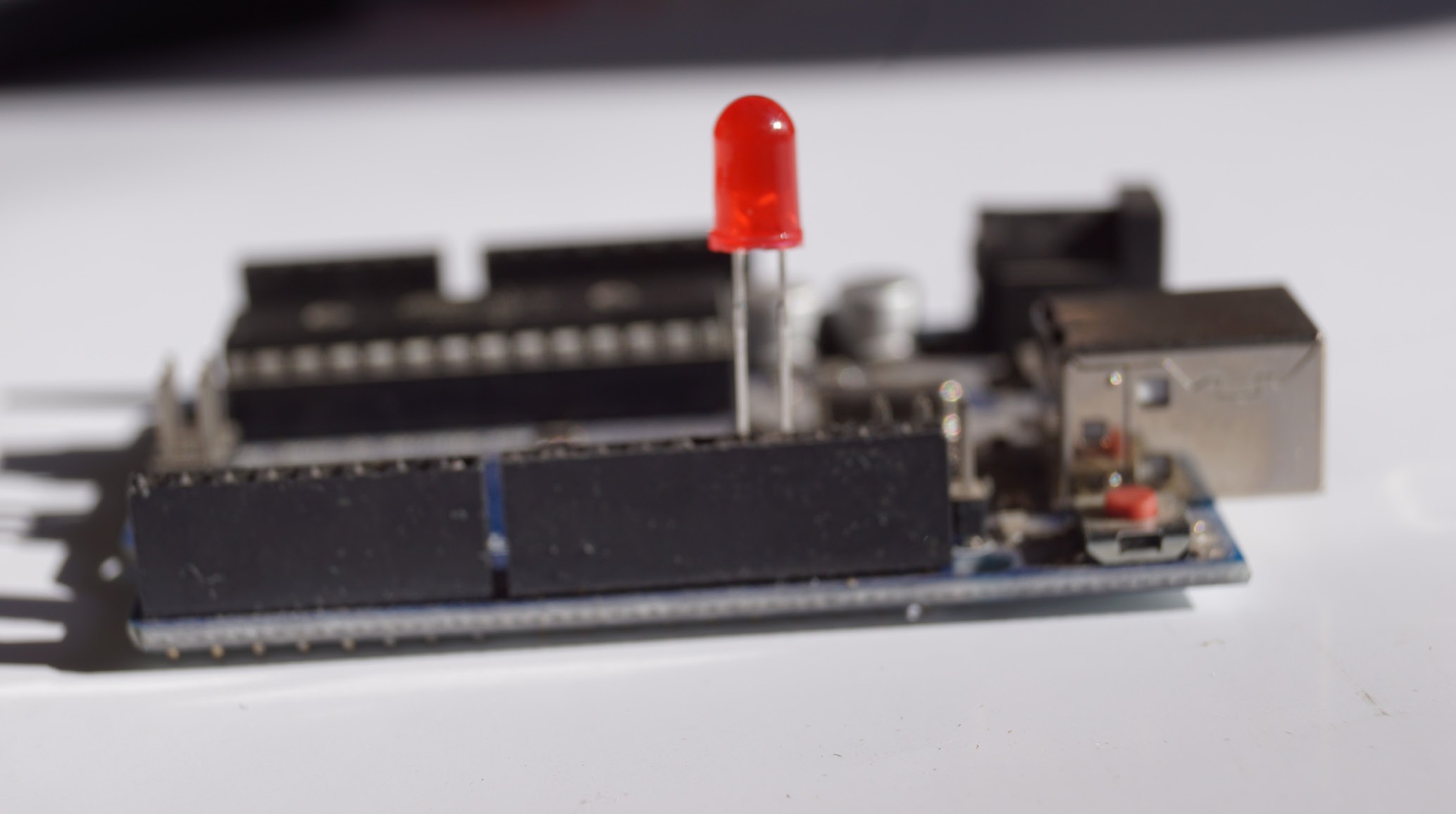
void Prende\_Led() { for (n=0;n<8;n++) { digitalWrite (leds[n], HIGH); delay (tiempo);

digitalWrite (leds[n], LOW);

delay (tiempo);

}

}

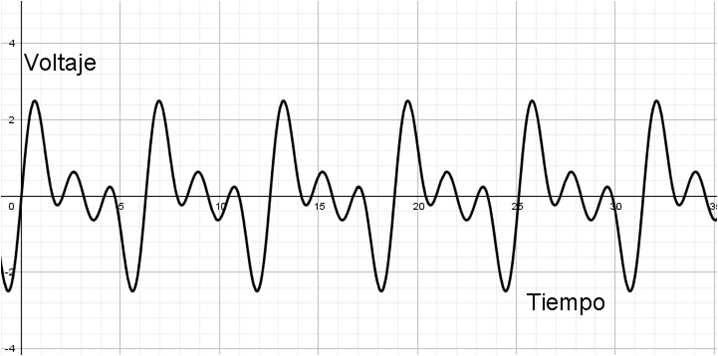


**6.- Encender Led con PWM**

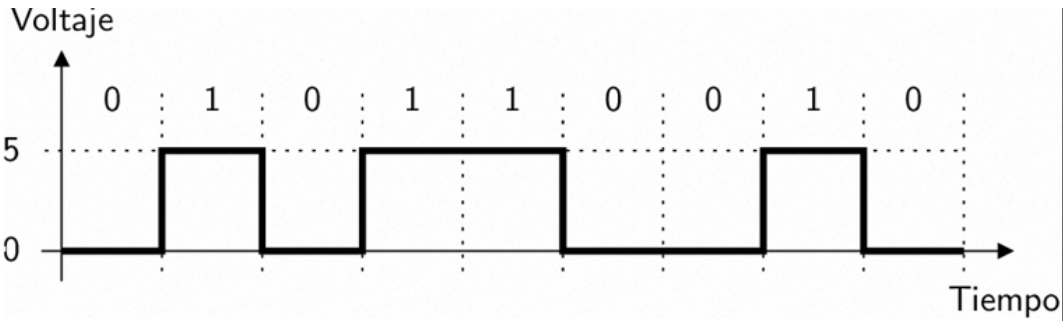
Encenderemos ahora los led pero por PWM ( modulación por ancho de banda o de pulso). Recordemos primero lo que significa una salida digital y analógica.

Una salida digital solo admite dos valores, en la placa Arduino es 0 V o 5 V. Una salida digital admite todos los valores posibles entre 0 V y 5 V.

Ejemplo de señal Analógica



Ejemplo de Señal Digital



Como la placa Arduino dispone de salidas digitales, la emulación de una señal digital a una analógica se llama

Modulación por ancho de pulso.

Tema que será tratado más adelante más detenidamente, cuando veamos por ejemplo el tema de los motores. Donde su velocidad dependerá de esta variable, al igual si queremos prender una luz secuencial.

Los pines que tienen esta opción son aquellos que van con la figura (~)

Si miramos la figura 6, vemos que los pines 3, 5 y 6 tienen esta figura(~), esa figura significa que el pin puede emular una salida analógica.

Utilizando la misma configuración de la experiencia anterior, vamos a hacer que los LED prendan en forma secuencial, pero solo en los pines 3, 5 y 6 tienen o emulan una salida analógica, los pines 2, 4 y 7 son digitales,



por lo tanto veremos que solo los pines con salida analógica serán secuenciales; los conectados a los pines digitales tiene dos opciones: o prenden o están apagados.

Para este ejemplo vamos a probar las salidas analógicas y digitales de la placa Arduino.

Si miramos la figura de la placa Arduino, vemos que los pines 2,3,4,5,6,7 algunos tienen un signo “~”

y otros no tienen nada.

Vemos que los pines 3, 5 y 6 tienen esta figura(~), esa figura significa que la salida emula una salida analógica.

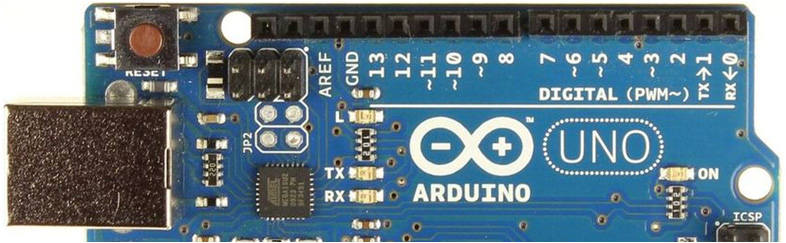


Fig. 6

**Recordemos algunos conceptos**

Un byte es una unidad de información formada por una seguidilla de bits adyacentes. Con el tiempo, se fijó el tamaño de un byte en 8 bits y se declaró como un estándar a partir de IBM S/360. ( ISO/IEC 80000-13)

1 Byte = 8 bits

1 bits es la cantidad de información más básica en electrónica y significa dos situaciones muy básica, si por un conductor pasa o no pasa corriente, que matemáticamente se expresa como uno o cero (1 o 0)

Por lo tanto un Byte tendrá la cantidad de bits de: 2 elevado a 8: 2\*2\*2\*2\*2\*2\*2\*2 = 256

Por qué 255 ?; porque es 2 elevado a 8 = 256, considerando el cero como primer valor tenemos 255. Para ello haremos el siguiente programa:

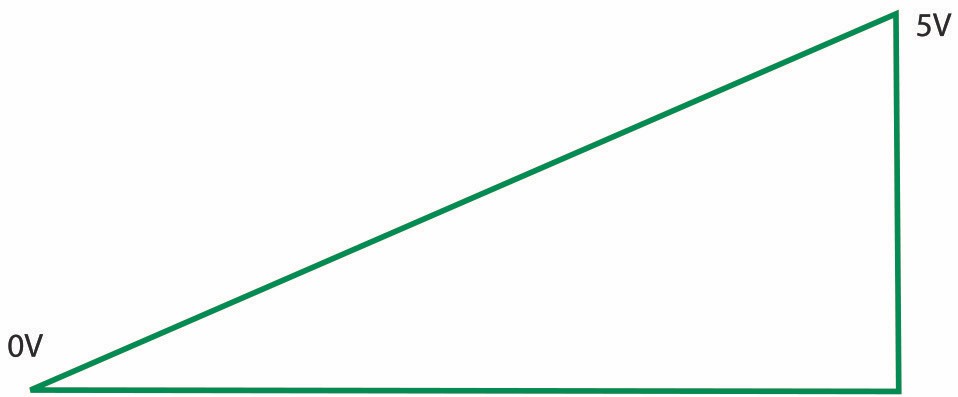
Vamos a dividir la salida entre 0 V y 5 V en 255 partes

Tomemos el siguiente ejemplo:

Imaginen que estamos en un primer piso de una casa y necesitamos subir al segundo piso, para eso voy a poner una escalera, y esta escalera tendrá varios escalones, una escalera en promedio tiene 14 escalones pero,

¿qué pasaría si ponemos 255 escalones?.

Mientras más escalones tenga la escalera, esta más se parecerá a una rampa .



Entonces vamos a dividir entre los 0 V y los 5 V entre 255 pasos; es decir, el ciclo For varia entre 0 y 255.

Vamos a incrementar los valores desde 0 a 255 y veremos que solo los conectados a la salida analógica variarán su brillo, el resto solo tendrá dos opciones: o se prende o está apagado.

Luego decreceremos, es decir pasamos de una señal de 5 V a 0 V dividido en 255 pasos

#define LED 2

#define LED1 3

#define LED2 4

#define LED3 5

#define LED4 6

#define LED5 7

void setup() { pinMode(LED,OUTPUT); pinMode(LED1,OUTPUT); pinMode(LED2,OUTPUT); pinMode(LED3,OUTPUT); pinMode(LED4,OUTPUT); pinMode(LED5,OUTPUT);

}

void loop() {

for(int i = 1; i <= 255; i++){ analogWrite(LED,i); // pin2 digital analogWrite(LED1,i); // pin3 análogo analogWrite(LED2,i); // pin4 digital analogWrite(LED3,i); // pin5 análogo analogWrite(LED4,i); // pin6 digital analogWrite(LED5,i); // pin7 digital delay(50);

}

delay(50);

for(int i = 255; i >= 1; i--){ analogWrite(LED,i); // pin2 digital analogWrite(LED1,i); // pin3 análogo analogWrite(LED2,i); // pin4 digital analogWrite(LED3,i); // pin5 análogo analogWrite(LED4,i); // pin6 análogo analogWrite(LED5,i); // pin7 digital delay(50);

}

delay(50);

}

La importancia de conocer el funcionamiento de los led es para saber si nuestra placa Arduino esta haciendo lo que indicáramos a través de un programa que hiciera.

Los led vendrían a ser los periférico de salida de nuestra placa Arduino .

Si estoy midiendo distancia o midiendo cantidad de gas, el Arduino a través de los led me indicarán si cumple la restriccion de distancia así como indicarme si hay determinada concentración de gas en un determinado ambiente, y se complementa perfecta con otros elementos como los Buzzer, que frente a una señal de alarma tenemos el led y el sonido de advertencia.



**R'Dbdt11:a**



RECUERDOS DE FUTURO

[**www.comercialnahuel.cl**](http://www.comercialnahuel.cl)[**http://roboticatecnociencia.cl**](http://roboticatecnociencia.cl)

**Hugo Marchant** [**h.marchant@roboticatecnociencia.cl**](mailto:marchant@roboticatecnociencia.cl)

**+ 56 9 5394 1513**

**Claudio Araya** [**caraya@comercialnahuel.cl**](mailto:caraya@comercialnahuel.cl) [**caraya@roboticatecnociencia.cl**](mailto:caraya@roboticatecnociencia.cl)

**+56 9 5199 7590**

**Amunátegui 31 Of. 40 Santiago**

**2 2633 9295**

**2 2633 0234**