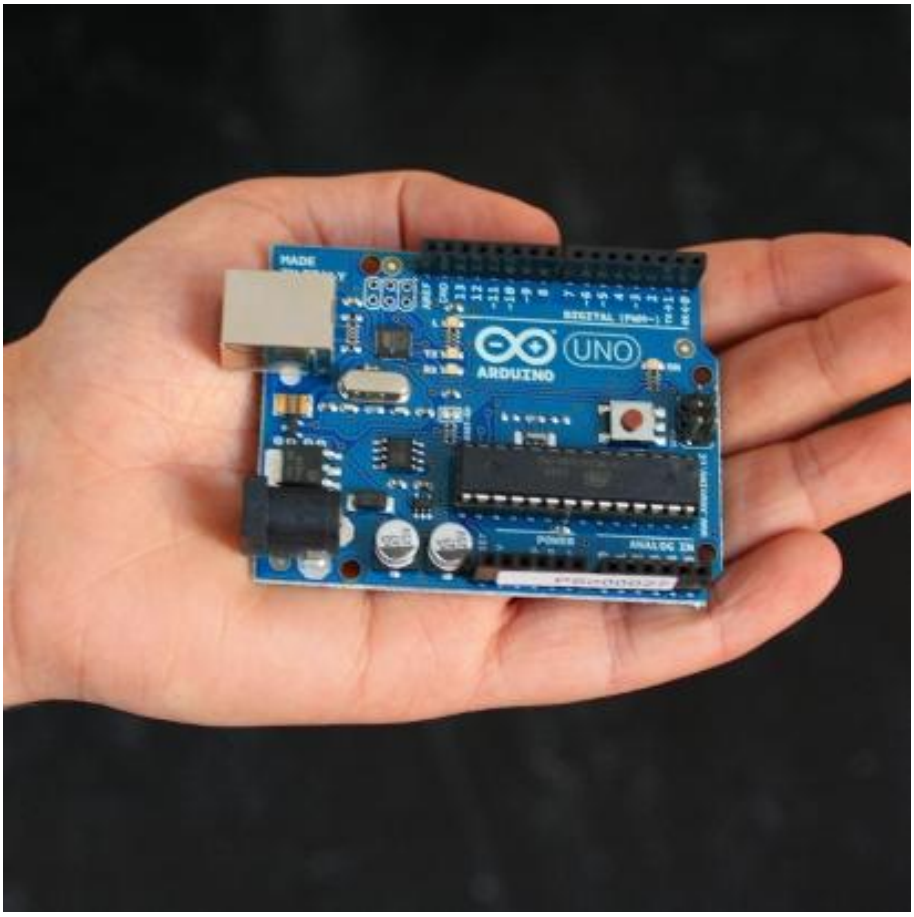
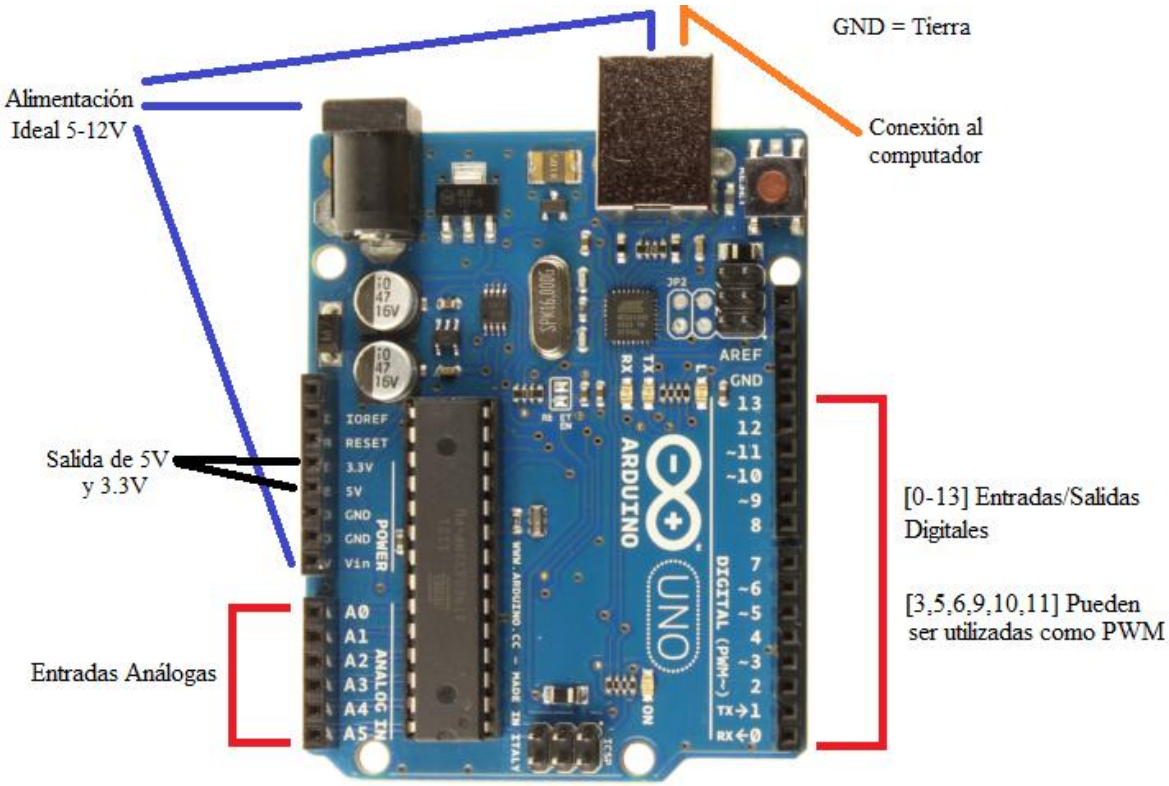
La Resistencia

# Taller #2

# “Introducción a Arduino”



Estructura Básica para realizar los primeros ejemplos, en el taller utilizamos el arduino duemilanov ATmega 328, pero los pines son los mismos.
La interfaz se baja de http://arduino.cc/en/Main/Software y seguir las instrucciones de http://arduino.cc/en/Guide/Windows.



## Programación

1. Principal

```
void setup()
{
 /*Aquí se anotan como va a iniciar el arduino, por ejemplo definir los
  pines que son salidas y los que son entradas*/
}

void loop()
{
  /*Aquí se escribe el programa*/
}
```

2. Digital

Para definir si un pin digital es entrada o salida se ocupa el comando
pinMode(N°Pin, INPUT/OUTPUT)

```
Ej:     pinMode(9, INPUT);          //Para Entrada
        pinMode(10, OUTPUT);        //Para Salida
```

Para escribir en la salida digital se ocupa el comando digitalWrite(N°Pin, HIGH/LOW)

```
Ej:     digitalWrite(10, HIGH);     //Para 1
        digitalWrite(10, LOW);      //Para 0
```

Para leer la entrada digital se ocupa el comando digitalRead(N°Pin)

```
Ej:     digitalRead(9);
```

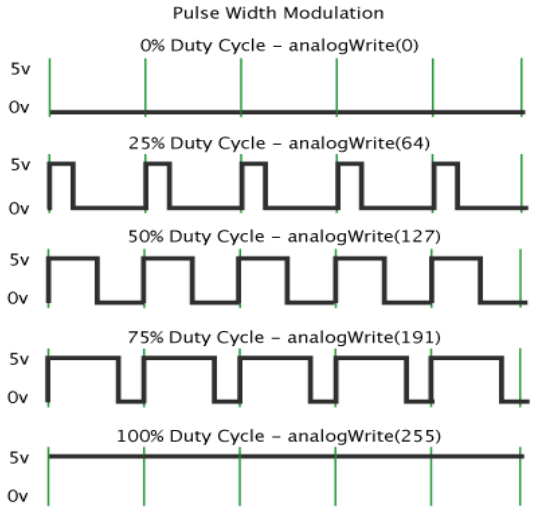
3. Análogo

Para utilizar las entradas análogas, no es necesario definir en el void setup() que es entrada. Para leer los datos se utiliza el comando analogRead(N°Pin), hay que tener en cuenta que los datos leídos están en el rango [0-1024].

```
Ej:     analogRead(A0);
```

4. PWM (Pulse-Width Modulation)

Los pines [3,5,6,9,10,11] pueden ser utilizados como PWM. Para esto, al igual que la entrada analógica no se debe definir que es una salida en el void setup(), se usa el comando analogWrite(duty-cycle), donde el duty-cycle o ciclo de trabajo es definido en arudino por la siguiente tabla.




Ej: analogWrite(130);

5. Extras

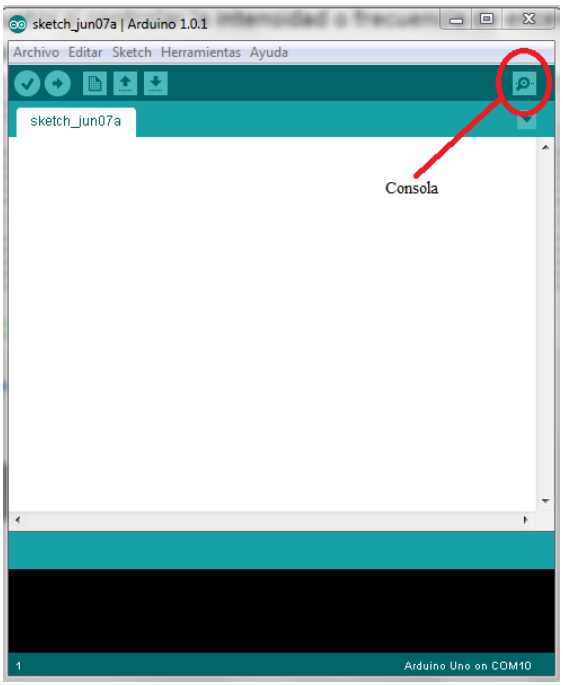
Para que el programa espere un tiempo antes de ejecutar la siguiente acción se puede utilizar el comando delay(tiempo_en_milisegundos), por ejemplo para esperar 1,5 segundos -> delay(1500);

Para cambiar de rango una variable se puede ocupar el comando map(variable_a_escalar, rango_inferior, rango_superior, rango_inferior_nuevo, rango_superior_nuevo), por ejemplo si ocupamos una entrada análoga, sabemos que su rango es [0-1024] y si necesitamos ocuparlas para cambiar el tiempo de encendido de un led y que varié entre 0.01s a 10s, entonces utilizaríamos
map(variable, 0, 1024, 10, 10000);

Siempre es bueno saber el valor de las variables cuando el programa se esta ejecutando, para esto hay que abrir el puerto serie

```
void setup() {
Serial.begin(9600);       //abre el puerto a 9600 bps
}
```

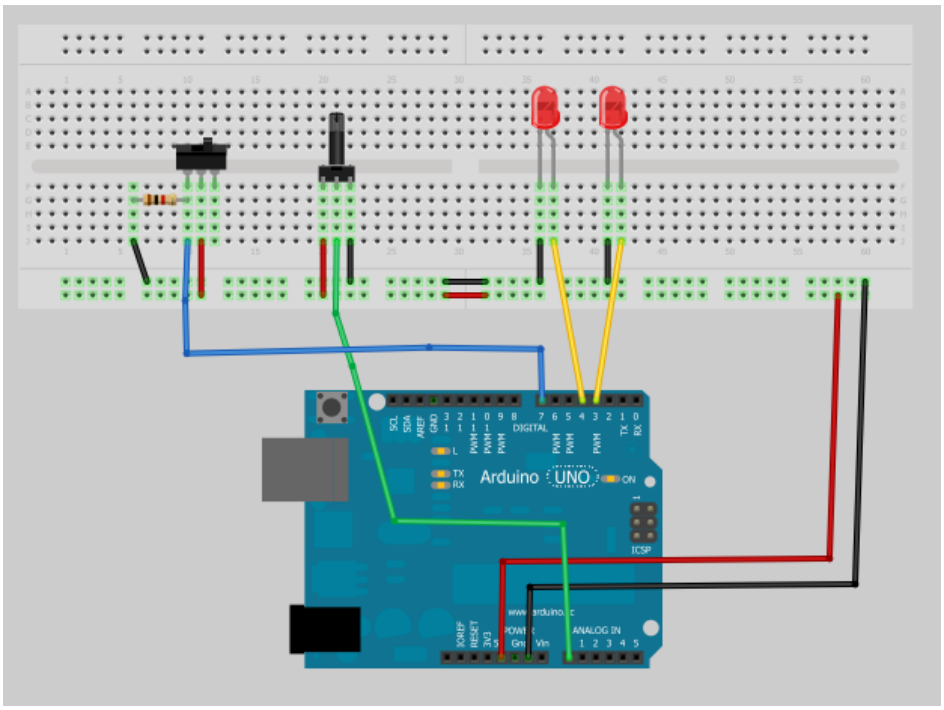
Luego escribiendo en el void loop() el comando Serial.print(lo_que_queremos_ver) podremos verlo en la consola. La consola se abre en la interfaz donde tiene una forma de lupa en la esquina superior derecha, se llama serial monitor.



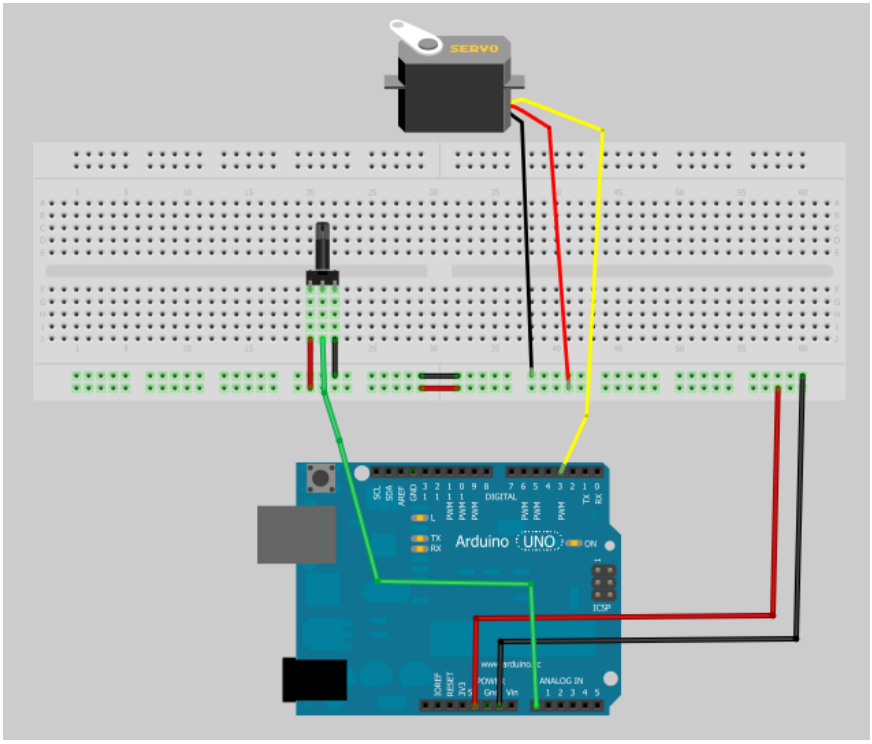

*En el siguiente link podrán ver todos los comandos y funciones que hay http://arduino.cc/es/Reference/HomePage , pero con esto es suficiente para realizar los primeros ejemplos del taller.

## Ejemplos

1. Elegir con un interruptor si controlar la intensidad o frecuencia de encendido de un led (Potenciometro de 10K- resistencia de 1K-2 Leds-1 interruptor)



2. Control de un Servo utilizando la librería
(servo-potenciómetro de 10K)




3. Controlar velocidad de un motor con un potenciómetro
(Transistor 2N2222-resistencia 1K-diodo 1N4148-potenciometro de 10K-Motor DC-batería para alimentar el motor)
*Mucho cuidado al conectar los cables