

PRÁCTICA VII

PROGRAMACION DE ENTRADAS ANALOGICAS

UTILIZACION DEL POTENCIOMETRO PARA SU CARACTERIZACION

Objetivo. Familiarizar al lector con la programación de las entradas Analógicas de la tarjeta Arduino, para su explicación utilizaremos el potenciómetro que acompaña a el laboratorio de Arduino.

Descripción. Como se menciona en la práctica anterior muchos de las entradas a la tarjeta Arduino no siempre son digitales es decir no siempre son del tipo on/off, muchas de las entradas son analógicas es decir provienen de señales que varían en el tiempo y que su valor no depende de un estado ALTO o BAJO sino de una serie de valores continuos, en esencia la Tarjeta Arduino puede procesar esta información utilizando los valores de 0 volts y 5 volts como los límites de voltaje en los que la señal analógica puede variar, en la figura 1 se muestra la forma de onda de una señal analógica.

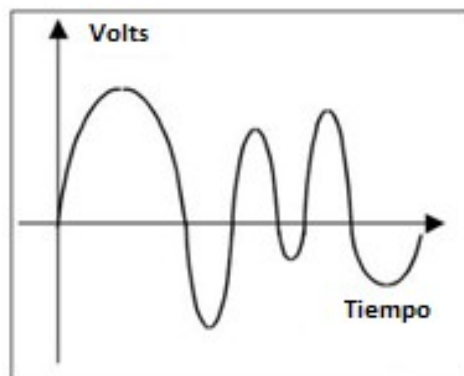


Figura 1.- Representación de una señal analógica

- La tarjeta Arduino es capaz de puede interpretar esta señal analógica y convertirla a valor digital "Convertidor analógico digital (ADC)" en un formato de 10 bits 2^9 , es decir la conversión analógica se realiza a través de mil 23 combinaciones de bits "de 0 a 1023", en consecuencia, podemos establecer el valor de la combinación 0 "000000000" para el 0 lógico correspondiente y el valor de 5 volts para la combinación 1023 (111111111).
- Si por ejemplo tenemos la lectura analógica de 512, entonces, estamos considerando que la señal analógica se encuentra justo a la mitad de su valor. Para clarificar este concepto consideremos el siguiente circuito, figura 2.

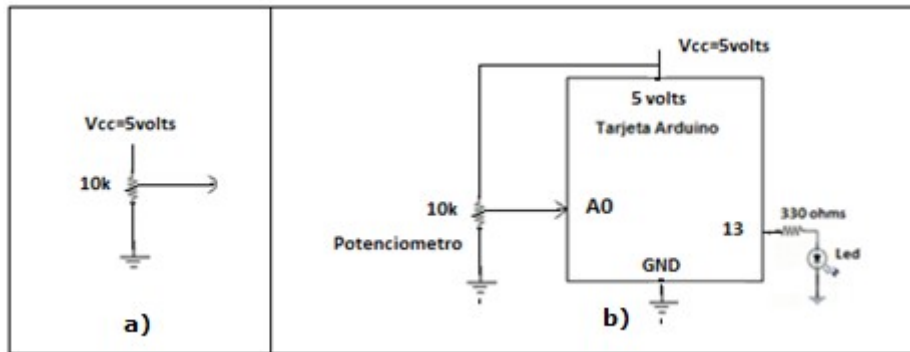


Figura 2.- Circuito para el análisis de entradas analógicas a Arduino.

- Como podemos observar en la figura 2a se ha colocado en los extremos de un potenciómetro de 10k, el voltaje de alimentación 5 volts y el voltaje GND respectivamente, el terminal central del potenciómetro lo conectamos a la entrada analógica A0 de la tarjeta Arduino que es donde se introducirá la variación de voltaje que simula la entrada analógica a la tarjeta, es decir el valor de la variación de voltaje en A0 depende de la variación del potenciómetro o resistencia variable.
- Si consideramos que la señal de entrada puede variar en un rango entre 0 y 5 volts equivalente a 0 y hasta 1023 combinaciones en formato de 10 bits como se mencionó anteriormente, entonces consideremos utilizar un voltaje intermedio, es decir un valor de 512 y podemos entonces establecer la siguiente condición.
- **Si la señal de entrada analógica es menor al valor 512 el led conectado en el pin 13 se mantendrá apagado y si la condición analógica es mayor que 512 entonces el led encenderá.** Para observar este comportamiento consideremos la siguiente programación de Arduino.

I PROCEDIMIENTO

1.1 Enunciado. Para esta práctica consideremos realizar el encendido y apagado de un led si el voltaje de entrada analógico es mayor o igual a 512 y se mantendrá apagado si el valor es menor a 512.

1.2 Inicie la aplicación de Arduino activando el símbolo del mismo.



1.3.- Realice la captura de código tal y como se muestra en la figura 3,

```
entradas_analogicas
1  /*PROGRAMACION ENTRADAS ANALOGICAS*/
2
3  //DECLARACION DE VARIABLES
4  int led=13;
5  int analogpin= A0;
6  int val=0;
7  int umbral=512;
8
9  // CONDICIONES INICIALES
10 void setup() {
11   pinMode(led, OUTPUT);
12 }
13 // PROGRAMACION
14
15 void loop() {
16   val=analogRead(analogpin);
17   if(val>=umbral){
18     digitalWrite(led,HIGH);
19   }
20   else
21   {
22     digitalWrite(led,LOW);
23   }
24 }
```

Figura 3. Código para manejo de la señal analógica

- En la línea 1 se describe el comentario de la práctica a realizar.
- A partir de la línea 3 iniciamos con la declaración de las variables involucradas.
- En la línea 4 se declara la variable entera `led` asignada al pin 13.
- En la línea 5 se declara la entrada analógica denominada en el ejemplo (**analogpin**) Entrada analógica ubicada en la entrada A0 de la tarjeta Arduino.
- En la línea 6 se declara el valor inicial de la entrada analógica `val=0`.
- En la línea 7 se declara el valor umbral que determina el funcionamiento de la entrada analógica “umbral = 512”
- **En la línea 11 se declara la salida led (led, OUTPUT) –es importante hacer notar que las entradas analógicas no se declaran en el void setup.**
- En la línea 15 iniciamos con el loop de programación
- La línea 16 debe de leerse como: Lee el valor analógico (**analogRead**) de la entrada **analogpin** y asígnalo a **val**, recordemos que inicialmente **val** se ha declarado con valor de 0, línea 6.
- Línea 17, dice, si **val** es mayor o igual que el voltaje de umbral 512, entonces escribe en la salida led un valor alto HIGH, línea 18, sino (else) línea 20, entonces asigna a la salida led un valor bajo “LOW” línea 22.

1.4 Para realizar la conexión en el Laboratorio de Arduino considere el siguiente procedimiento.

1.5 Conecte el voltaje de Arduino $V_{cc}=5V$ a la barra de voltajes $V_{cc}=5V$ al igual que su conexión de tierra GND a GND, figura 4.

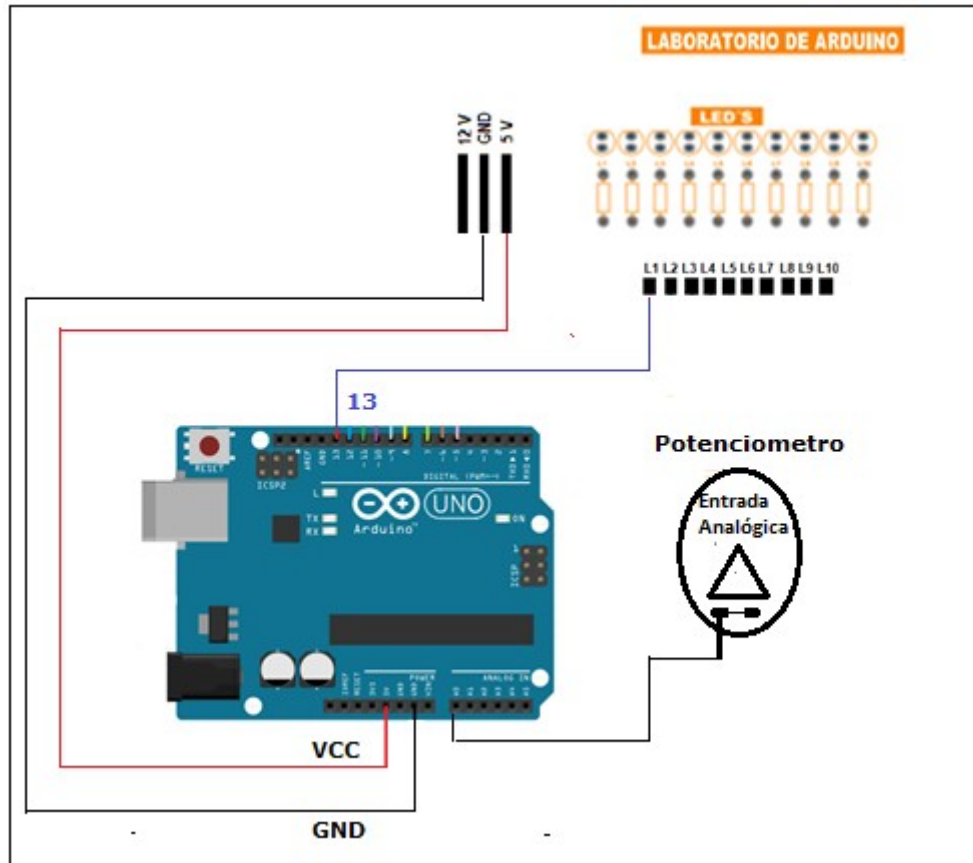


Figura 4. Conexión a la tarjeta Arduino

1.6 Conecte la entrada analógica a la entrada A0 y finalmente conecte la salida 13 al led L1.

1.7 Para probar el funcionamiento del circuito mueva el potenciómetro y observe el encendido y apagado del led L1.

2 Ejercicio

Realice un programa que le permita observar en el display de 7 segmentos el mensaje de HOLA, cuando la entrada analógica se encuentre en un umbral mayor a 640, figura 5.

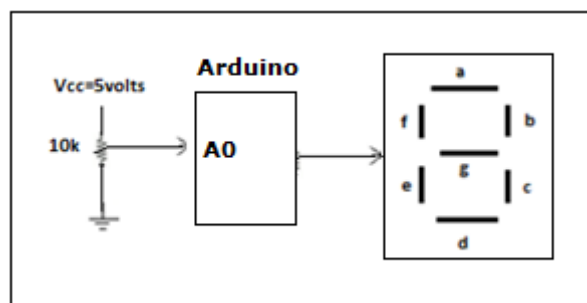


Figura 5. Ejercicio