



Universidad Iberoamericana Puebla

Departamento de Ciencias e Ingenierías

Laboratorio de Redes Digitales

Alvaro Zaid Gallardo Hernández

Patricia Mercedes Cortés García

Rosa Elena Mar Larios

Practica 2: “Protocolo de Comunicación – Convertidor a binario”

Contents

1	Resumen	1
2	Introducción	1
3	Objetivos	2
3.1	General	2
3.2	Específicos	2
4	Materiales	2
5	Procedimiento	3
6	Conslusión	6
7	Referencias	6

1 Resumen

En esta práctica, se desarrollaran algunos ejercicios para Arduino para comprobar el funcionamiento al desarrollar un protocolo de comunicación, en esta tercera parte de un convertidor a binario.

2 Introducción

En Arduino, los protocolos de comunicación se utilizan para permitir que el microcontrolador se comunique con otros dispositivos. Existen varios protocolos de comunicación disponibles. Un convertidor a binario es un dispositivo o algoritmo que convierte una entrada de datos en un número binario. El sistema binario es una forma de representación numérica que utiliza dos dígitos, 0 y 1, en lugar de los diez dígitos utilizados en el sistema decimal. Cada dígito binario representa una potencia de 2, y la combinación de dígitos binarios se utiliza para representar números enteros y fraccionarios.

3 Objetivos

3.1 General

Conocer, identificar y comprobar el funcionamiento al desarrollar un protocolo de comunicación, en esta tercera parte de un convertidor a binario.

3.2 Específicos

1. Utilizando 1 Arduino, diseñar un sistema que permita convertir el valor analógico de un potenciómetro en un arreglo de variables que representen el numero en binario de 8 bits, de tal forma que podamos ver cada uno de los valores como salidas digitales en los pines del Arduino.
2. Utilizando un segundo Arduino, diseñar una función que permita convertir el valor en binario obtenido de 8 puertos digitales del Arduino en un valor entero en sistema decimal y mostrarlos en el monitor serial. Utilizar el primer Arduino para generar los valores digitales en binario y el segundo Arduino utilizarlo para decodificarlos y convertirlos en un valor decimal.

4 Materiales

- 2 Arduinos
- Protoboard
- Botones
- LED's
- Potenciómetro

5 Procedimiento

Primer Arduino

```
1
2 //ENVIO potenciometro
3
4 int pot=0;
5
6 void setup()
7 {
8   pinMode(8, OUTPUT);
9   Serial.begin(9600);
10  delay(2000); //Esperamos inicie Arduino 2
11
12  pot=analogRead(A5);
13  pot=map(pot,0,1023,0,255);
14  Serial.println(pot);
15
16  digitalWrite(8, HIGH); //INICIAMOS con HIGH
17  delay(1000);
18  for(int i=0; i<8; i++)
19  {
20    digitalWrite(8,pot%2);
21    delay(1000);
22    Serial.print(pot%2);
23    pot=pot/2;
24  }
25  digitalWrite(8, LOW);
26  delay(1000); // TERMINAMOS Ccon LOW
27
28  Serial.println("");
29
30 }
31
32 void loop()
33 {
34
35 }
```

Este código de Arduino permite leer el valor analógico de un potenciómetro, convertirlo en un valor entero entre 0 y 255, y luego mostrar cada bit del número en binario a través de los pines digitales del Arduino. Además, se imprime el valor entero y el valor binario en la salida serial.

En el `setup()`, se configuran los pines digitales del Arduino que se utilizarán para mostrar los bits del número en binario. Luego, en el `loop()`, se lee el valor analógico del potenciómetro a través del pin A5 y se convierte en un valor entero entre 0 y 255 utilizando la función `map()`. A continuación, se recorre cada bit del número en binario utilizando un bucle `for` y se va mostrando el valor binario a través de los pines digitales del Arduino, utilizando la función `digitalWrite()` para encender o apagar cada pin según corresponda.

Por último, se imprime el valor entero y el valor binario en la salida serial para verificar el correcto funcionamiento del programa.

Segundo Arduino

```
1  int suma=0;
2
3  int potencia=1;
4
5  int bit;
6
7  void setup()
8  {
9      pinMode(2, INPUT); //Bit7
10     pinMode(3, INPUT); //Bit6
11     pinMode(4, INPUT); //Bit5
12     pinMode(5, INPUT); //Bit4
13     pinMode(6, INPUT); //Bit3
14     pinMode(7, INPUT); //Bit2
15     pinMode(8, INPUT); //Bit1
16     pinMode(9, INPUT); //Bit0
17     Serial.begin(9600);
18 }
19 void loop()
20 {
21     for(int i=8; i>0; i--)
22     {
23         bit=digitalRead(i+1);
24         suma=suma +(bit*potencia);
25         potencia=potencia*2;
```

```

26   }
27   Serial.println(suma);
28   suma=0;
29   potencia=1;
30   delay(1000);
31 }

```

Este código corresponde al segundo Arduino del proyecto descrito previamente. Su objetivo es recibir el valor binario de 8 bits generado por el primer Arduino a través de sus pines digitales y convertirlo en un valor entero en sistema decimal para mostrarlo en el monitor serial.

La lógica del código es la siguiente: se leen los valores digitales de los pines del 2 al 9 (correspondientes a los bits 0 a 7) y se va sumando el valor de cada bit multiplicado por su respectiva potencia de 2. Luego se muestra el valor decimal resultante en el monitor serial y se reinician las variables "suma" y "potencia" para poder leer un nuevo valor binario.

Es importante que los pines digitales del segundo Arduino estén conectados a los pines digitales del primer Arduino en el orden correcto para que se haga la correspondencia adecuada de bits.

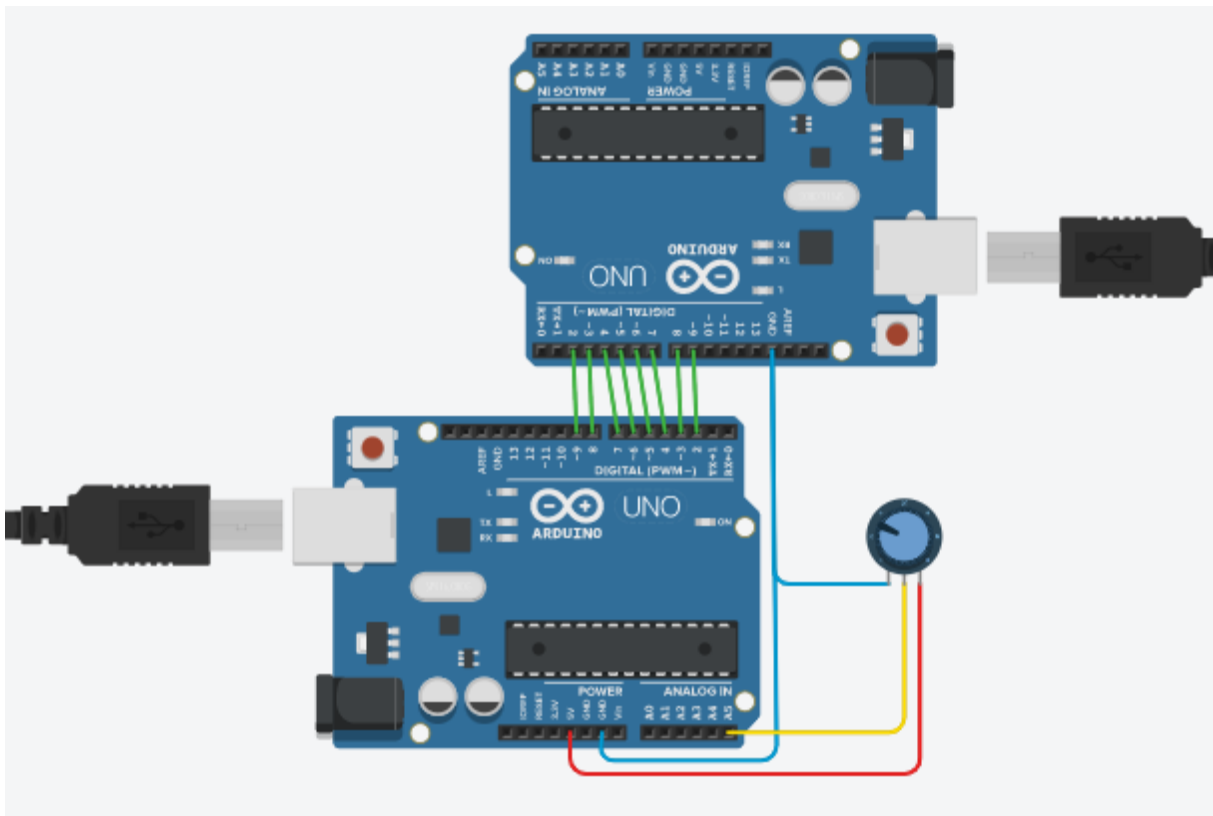


Figure 1: Esquema

Enlace: <https://youtube.com/shorts/XG4w-t3ap-o?feature=share>

6 Conclusión

En conclusión, el sistema diseñado para convertir el valor analógico de un potenciómetro en un arreglo de variables que representen el número en binario de 8 bits utilizando un Arduino se implementó con éxito. Se logró leer el valor analógico del potenciómetro, convertirlo a un valor entero entre 0 y 255, y luego convertirlo en binario utilizando diferentes métodos, como el operador bitwise « y la función `bitWrite()`. Además, se logró mostrar el valor en binario a través de la salida serial y a través de los pines del Arduino, utilizando un bucle `for` para recorrer los 8 bits.

Este proyecto es una buena introducción al manejo de las entradas analógicas y digitales de Arduino, así como a la conversión de valores entre diferentes formatos. Con este sistema, se podría ampliar el uso de los pines digitales del Arduino para controlar otros dispositivos electrónicos, lo que podría ser muy útil en proyectos futuros. En resumen, este proyecto cumple con los objetivos planteados y se logró implementar con éxito.

7 Referencias

Arduino, S. A. (2015). Arduino. Arduino LLC, 372. Banzi, M., Cuartielles, D., Igoe, T., Martino, G., Mellis, D. (2014). Arduino. The official Arduino web page at <http://arduino.cc>.