

## **Bitacora del primer proyecto de Fundamentos de Arquitectura de Computadores**

**Nombre:** Daniel Serrano Cañas

**Carnet:** 2019031631

**Curso:** Fundamentos de Arquitectura de Computadores

**Professor:** Luis Alberto Chavarría Zamora

**Instituto:** Instituto Tecnológico de Costa Rica

### **Bitácora**

**16/09/2025**

Este día se inicia la con el diseño del sensor para la entrada del circuito. Para este se decide usar un piezoelectrico debido a su bajo costo y simplicidad, ya que este al recibir algún tipo de vibración genera un pulso en CA. Sin embargo, este pulso era demasiado pequeño, por lo que se opta por amplificar este con un arduino para que diera 5V usando un pin digital, cada vez que detectara el pulso del piezoelectrico.

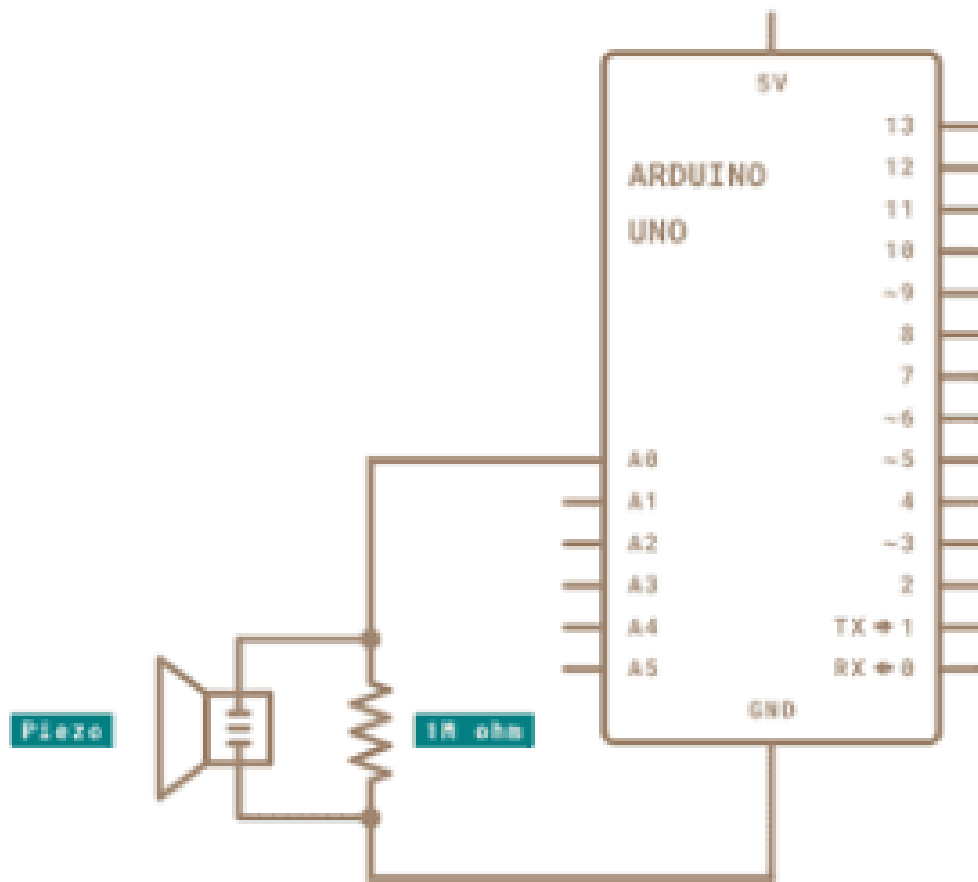


Figure 1: Circuito piezoelectrico, tomado de Arduino Docs: Detect a knock

Usando este circuito como base, se usó una resistencia de  $2k\ \Omega$  entre el pin A0 y GND, para filtrar la señal y evitar ruido. Dentro del arduino, se le cargó un código sencillo para detectar el pulso a cierta intensidad y con base a eso, recibir la señal.

```
const int piezoPin = 8;
const int piezoPinIn = A0;
const int treshholdPiezo = 40;
void setup(){
  pinMode(piezoPin, OUTPUT);
}
void loop(){
  int sensorValue = analogRead(piezoPinIn);
  if (sensorValue > treshholdPiezo) {
    digitalWrite(piezoPin, HIGH);
  }
}
```

```
} else {  
    digitalWrite(piezoPin, LOW);  
}  
}
```

Con esto, se pudo lograr el circuito para sentir los golpes.

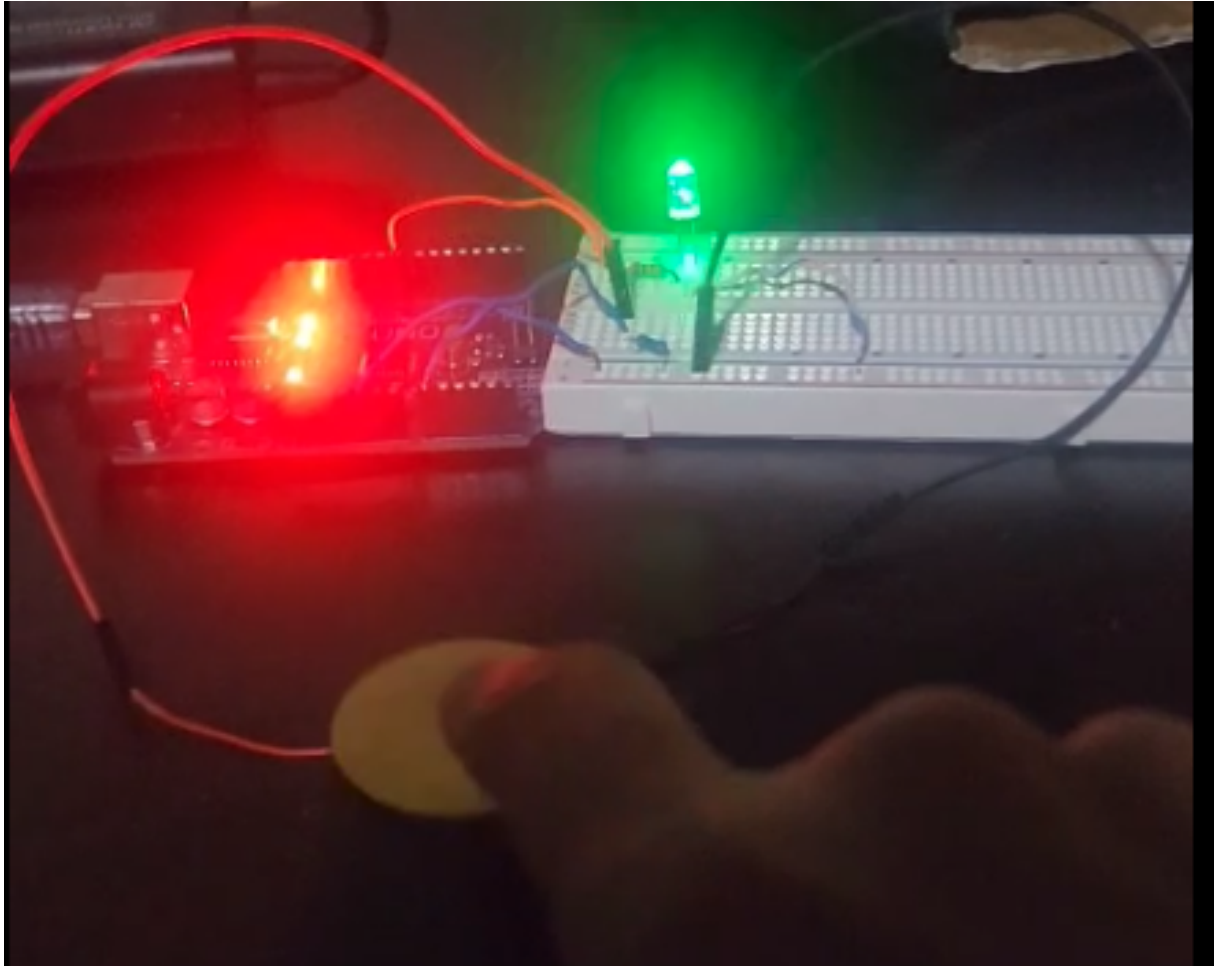


Figure 2: Circuito piezoelectrico detectando un golpe

**18/09/2025**

Debido a conflictos con la pareja, todo el taller se realizó en un día de forma individual. Para iniciar se definió una contraseña para abrir y otra para cerrar la puerta como se observa en las tablas.

Table 1: Contraseña de abertura.

A	B	C	D	E	F	G	H	Abrir
1	0	0	1	1	1	0	1	1

Table 2: Contraseña de cerrado.

A	B	C	D	E	F	G	H	Cerrar
1	1	1	0	0	1	1	1	1

Con estas tablas podemos sacar los minterminos de cada una, ya que cada decoder va a hacer una accion diferente. Primeramente para la tabla 1 tenemos el mintermino

$$Abierto = (\overline{A}\overline{B}\overline{C}DE\overline{F}\overline{G}H)$$

Seguidamente para cerrar el mintermino que se obtiene con base a la tabla 2

$$Cerrado = (ABC\overline{D}\overline{E}FGH)$$

Con esto podemos armar el los circuitos correspondientes. Para el circuito de abertura tendríamos

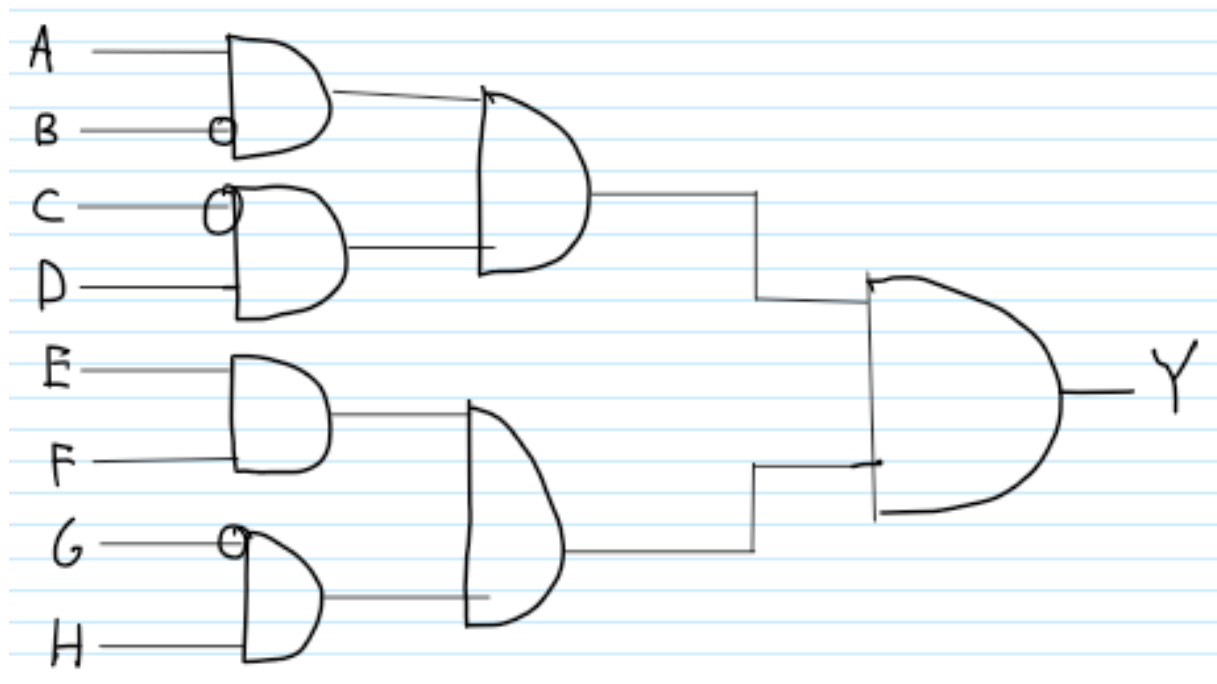


Figure 3: Circuito decodificador para abrir

Sin embargo, si analizamos, las entradas BC se pueden juntar, ya que siempre van a estar negadas para entonces usar una compuerta AND menos, por lo que el circuito se simplifica a

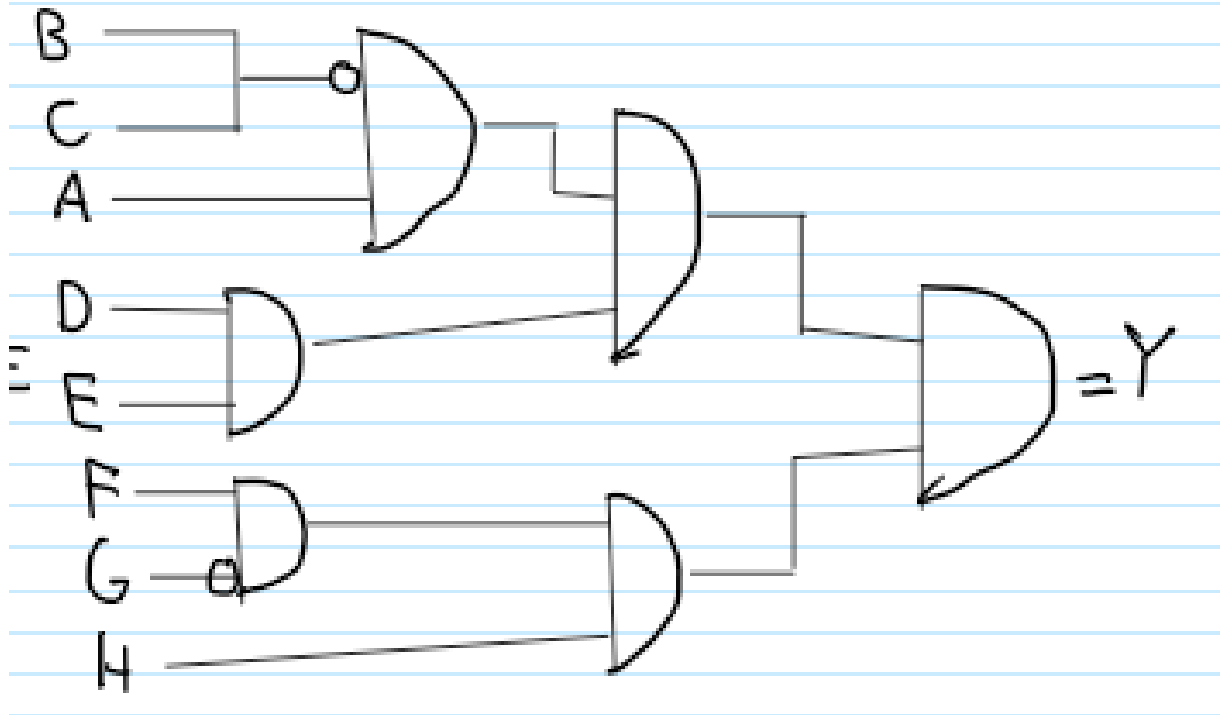


Figure 4: Circuito decodificador para abrir simplificado

Ahora para el circuito para decodificar la contraseña de cierre, con base al mintermino sacado de la tabla 2, tenemos

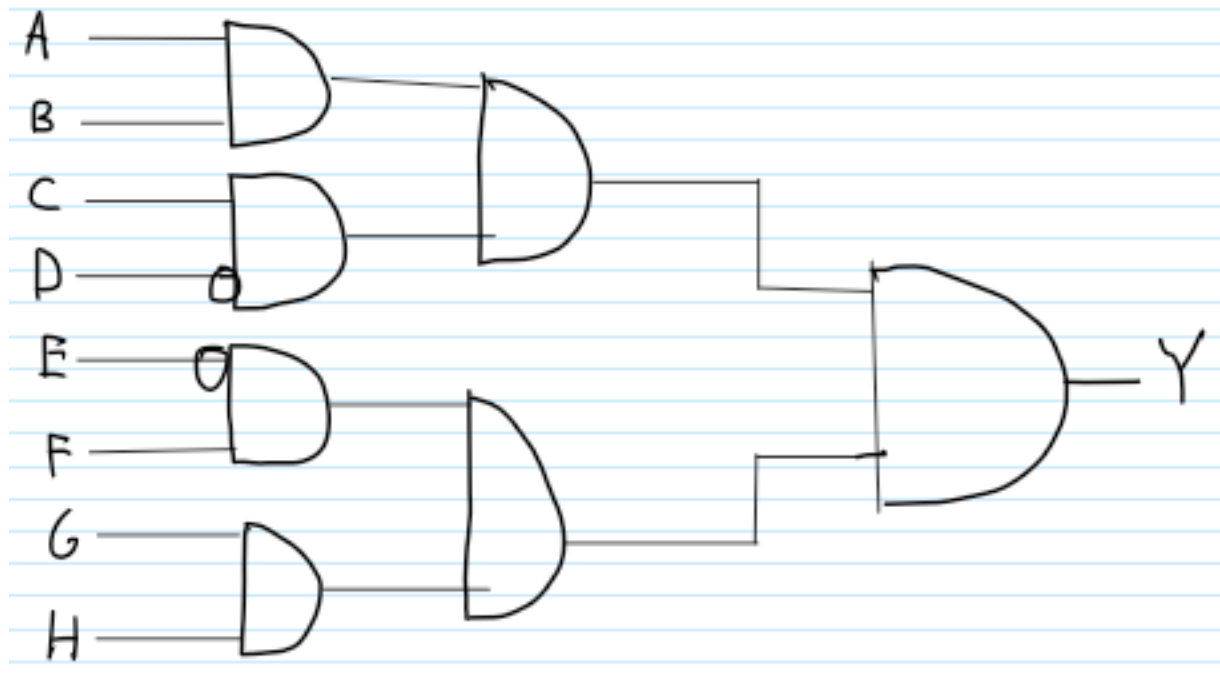


Figure 5: Circuito decodificador para cerrar

Sin embargo, si analizamos, las entradas DE se pueden juntar, ya que siempre van a estar negadas para entonces usar una compuerta AND menos, por lo que el circuito se simplifica a

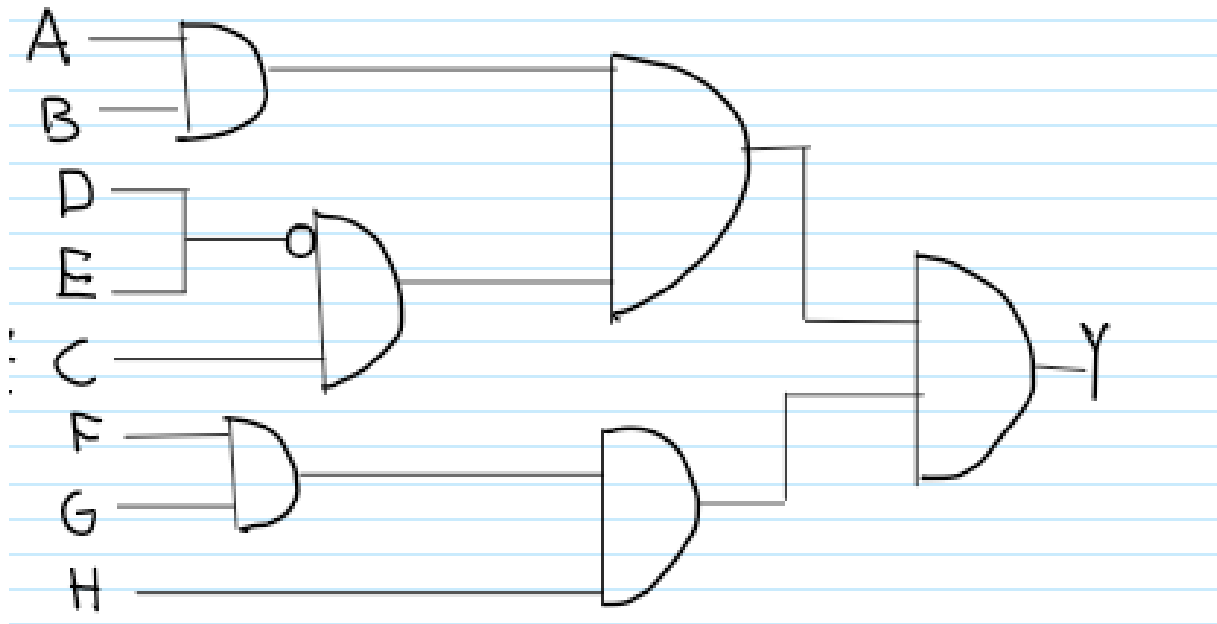


Figure 6: Circuito decodificador para cerrar simplificado

Una vez obtenido esto, ahora tenemos dos salidas, la del decodificador de apertura (la llamaremos A a partir de ahora) y la del decodificador de cerrado (la llamaremos B a partir de ahora).

Ahora lo que se busca es que solo cuando uno de estas dos entradas es 1, se abra el circuito. Analizando las tablas de algebra booleana, podemos identificar que la tabla del XOR es la que sirve para esto, por lo que será la principal compuerta que tendremos en mente para los segmentos del display de 7 segmentos que usaremos. Ahora, hay que definir los segmentos que se van a encender, esto se hizo con base a la siguiente imagen.

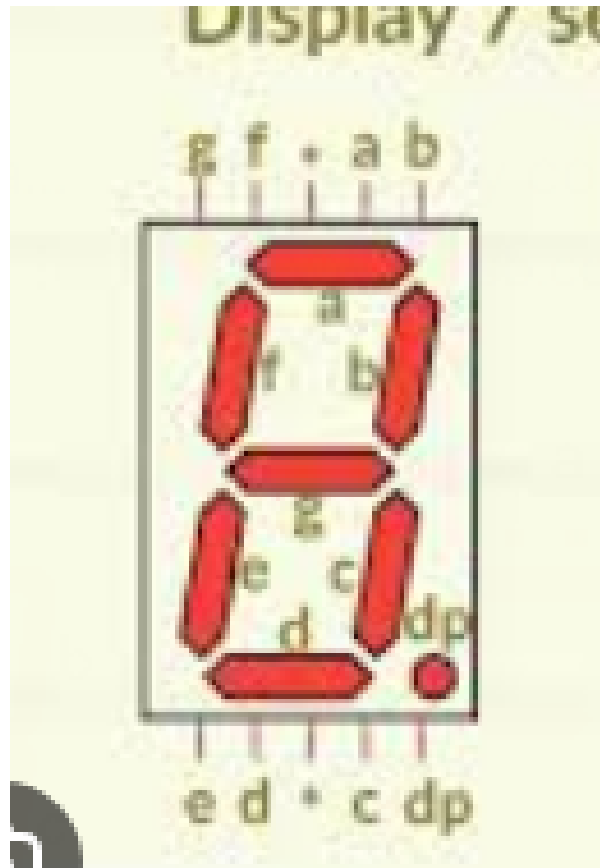


Figure 7: Diagrama display 7 segmentos

Entonces, analizando 7, para que una A de abierto ocupamos que se enciendan los segmentos a, f, b, e, c y g. Y para en el caso de la C de cerrado, ocupamos que se enciendan los segmentos a, f, e y d. Sabiendo esto, podemos construir una tabla para analizar esto.

Table 3: Análisis de los 7 segmentos deseados a encender.

A	B	a	b	c	d	e	f	g	XOR
1	0	1	1	1	0	1	1	1	1
0	1	1	0	0	1	1	1	0	1

Si se analiza la tabla 3, podemos notar que en el caso que el XOR sea 1, siempre se encienden los segmentos a, e y f. Por lo que podemos conectar A y B con un XOR para encender a, e y f. Ahora, para los demás, no es tan intuitivo, pero si se analiza, b, c y g, se encienden en la misma condición, por lo que se pueden agrupar y analizar su mintermino.

Table 4: Mintermino segmentos b, c y g

A	B	Sal	Mintermino
1	0	1	$A\bar{B}$
0	1	0	-

Seguidamente, analizando la tabla 3, se realiza el mintermino para la salida que falta, la de d, esto sería entonces

Table 5: Mintermino segmentos d

A	B	Sal	Mintermino
1	0	0	-
0	1	1	$B\bar{A}$

Con esto, podemos armar un circuito combinatorio, que dependiendo de las salidas de los decodificadores, mostrará una C o una A en el display de 7 segmentos.

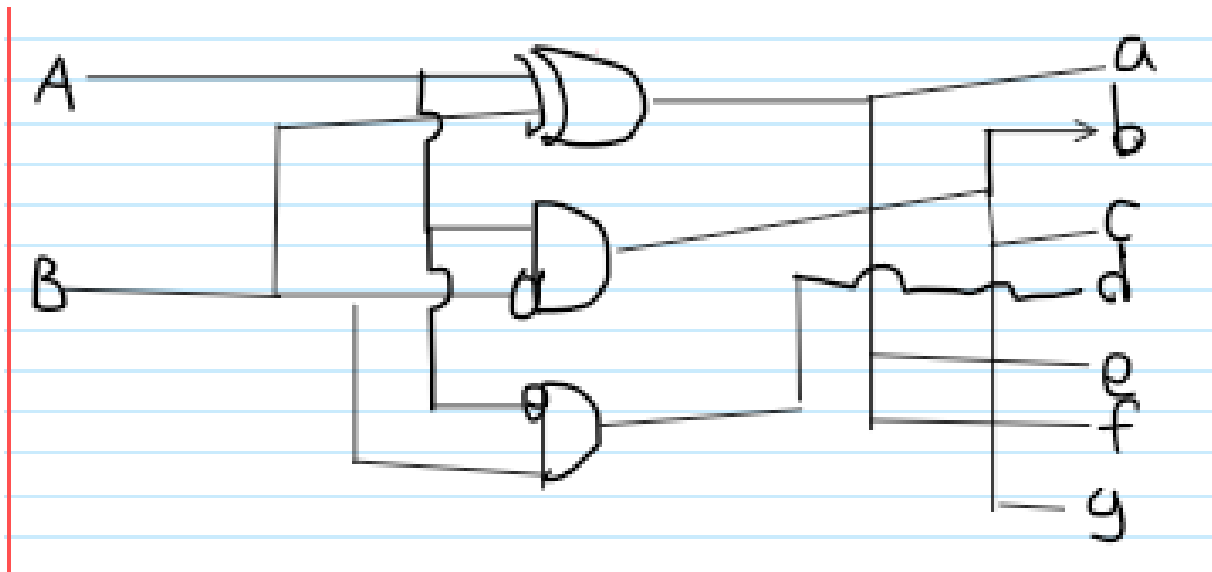


Figure 8: Circuito combinatorio para el display de 7 segmentos

Con esto, los circuitos de las figuras 4, 6 y 8, al juntarse de en un simulador como tinkercad, se ve de la siguiente forma



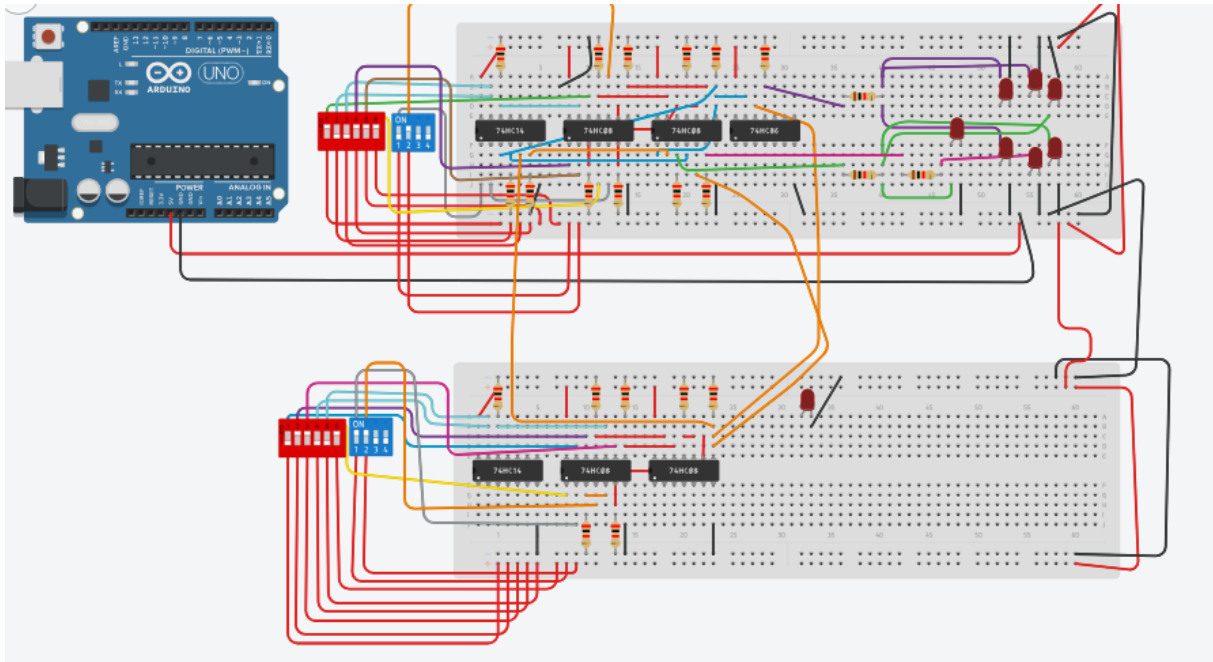


Figure 9: Circuito completo en tinkercad

En la figura 9 se observan 2 sets de switches para ambos decoders, pero es para efectos de prueba, se pueden juntar conectando ambos decoders en paralelo y la salida, al final se conecta a un único circuito combinatorio que enciende el decoder. En la figura 9 también se puede observar que se usaron leds, esto debido a que el display de 7 segmentos de Tinkercad tiene los diodos internos invertidos, por lo que se usaron los leds para simular el display de 7 segmentos que se usará para la demostración física del circuito.

El circuito final montado se aprecia en la siguiente imagen.

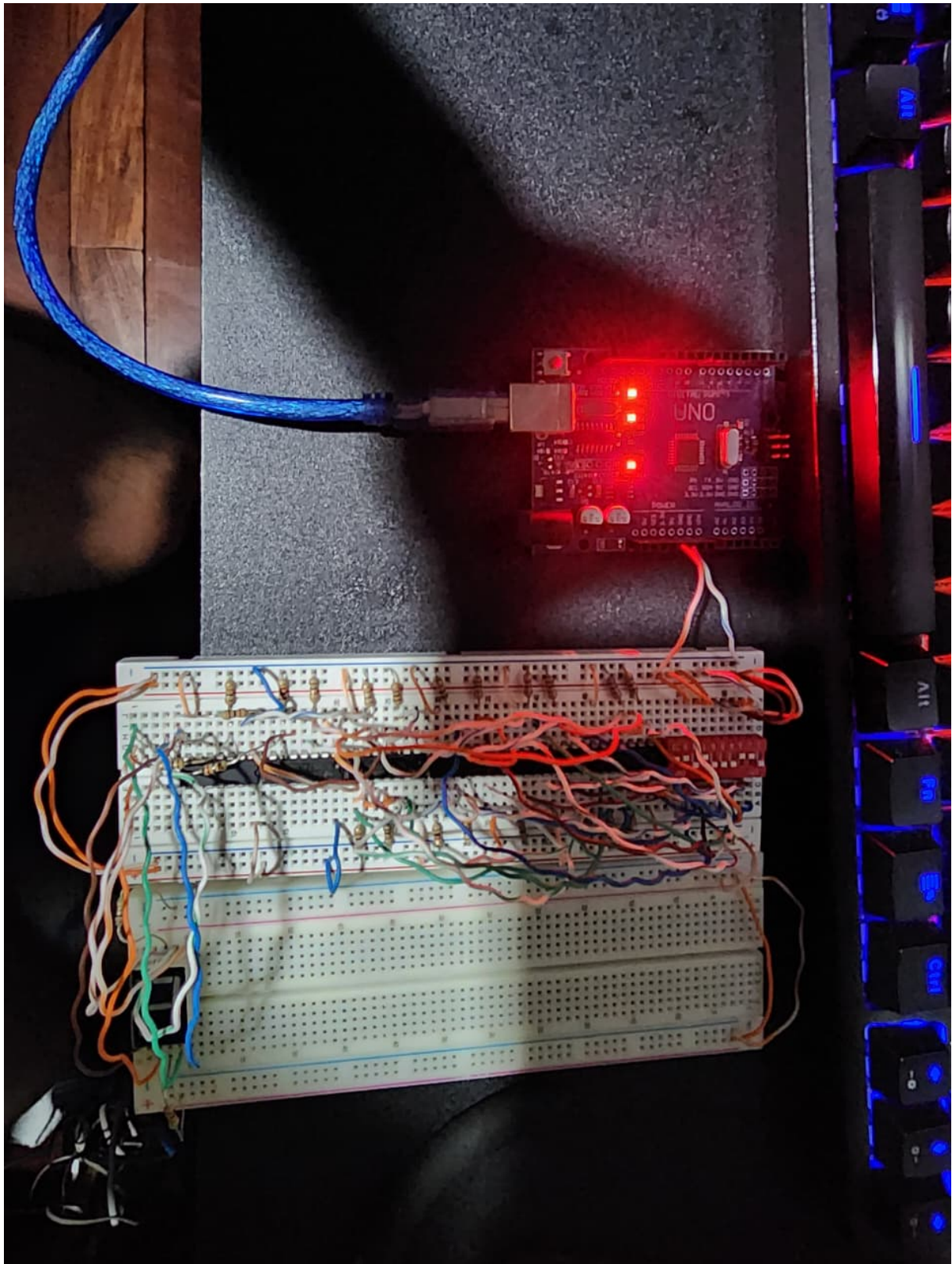


Figure 10: Circuito final fisico