

Práctica - Sensores I



Daniela Alejandra Córdova Porta

Richard Junior Mercado Correa

UNIVERSIDAD COMPLUTENSE DE MADRID

Madrid, diciembre de 2020

Resumen

Este documento expone el procedimiento y análisis realizado en la práctica 3 - Sensores I de la asignatura Robótica en la Universidad Complutense de Madrid. En primer lugar, se establecen los objetivos de la práctica y su fundamento teórico. Luego, se establece el procedimiento para realizarla; explicando las conexiones y código necesarios. Finalmente se discuten los resultados obtenidos y se exponen los errores cometidos.

Índice

Resumen.....	2
1. Objetivos	4
2. Fundamento Teórico	4
2.1. Resistencia variable con la luz (LDR)	4
2.2. Sensor digital (CNY70)	4
3. Procedimiento	5
3.1. Lectura de datos analógicos con la Raspberry Pi.....	5
3.2. Arquitectura software del robot	9
4. Resultados y Análisis de Resultados.....	12
4.1. Lectura de datos analógicos con la RaspBerry Pi.	12
4.2. Arquitectura software del robot	14
5. Discusión y conclusiones.....	14
5.1. Conclusiones:.....	14
5.2. Errores:.....	15
6. Bibliografía	15

1. Objetivos

- Conocer los elementos de entrada/salida: GPIO y wiringPi.
- Uso básico de sensores con la Raspberry Pi: lectura de sensores digitales.
- Diseño de programa para robot móvil: lectura de sensores a través de hilos.

2. Fundamento Teórico

Los sensores son dispositivos que permiten medir variables internas o externas del robot. Esto permite que el comportamiento de nuestro robot cambie según los valores detectados por estos sensores. Estos dispositivos pueden ser internos, si miden variables propias del robot, o externos, si detectan variables ajenas al robot. En esta práctica se utilizarán sensores externos de proximidad, y serán utilizados como elementos digitales.

Los sensores utilizados son:

2.1. Resistencia variable con la luz (LDR)

El LDR se trata de un dispositivo que varía su valor de resistencia eléctrica dependiendo de la luz que incide sobre él. El valor de resistencia de un LDR es bajo cuando hay una luz incidiendo sobre él y muy alto cuando no hay luz.

La variación del valor resistivo de un LDR tiene cierto retardo, que es diferente si se pasa de oscuro a iluminado o de iluminado a oscuro. Por esta razón un LDR no se puede utilizar en algunas aplicaciones, en especial en aquellas en que la señal luminosa varía con rapidez. El tiempo de respuesta típico de un LDR está en el orden de una décima de segundo. En este caso se utilizará el LDR como un sensor binario, para saber si hay luz suficiente o no.

2.2. Sensor digital (CNY70)

El CNY70 es un sensor óptico reflexivo de infrarrojos de corto alcance (de 0,3 a 10 mm) basado en un diodo emisor de luz infrarroja y un receptor formado por un fototransistor, ambos apuntando a la misma dirección. Su funcionamiento se basa en la capacidad de reflexión del objeto, y la detección del rayo reflejado por el receptor.

Tiene un alcance muy corto, por lo que no se suele utilizar para detectar obstáculos. Se utilizará este sensor para la detección y seguimiento de líneas.

El circuito implementado consiste en que cuando el sensor está sobre la línea negra, el sensor permanece en corte, devolviendo un valor muy alto a la salida en el colector, y si se encuentra sobre un fondo blanco, el haz infrarrojo se refleja y el fototransistor se satura entregando a la salida del circuito un nivel bajo.


```

10. wiringPiSetup () ;
11. int i;
12. pinMode (0, INPUT) ;
13. while(1){
14.
15.   i = digitalRead(0);
16.   printf("%d",i);
17.
18. }
19. return 0;
20. }

```

Este mismo imprimirá en pantalla 0 si no está tapado y 1 si lo está.

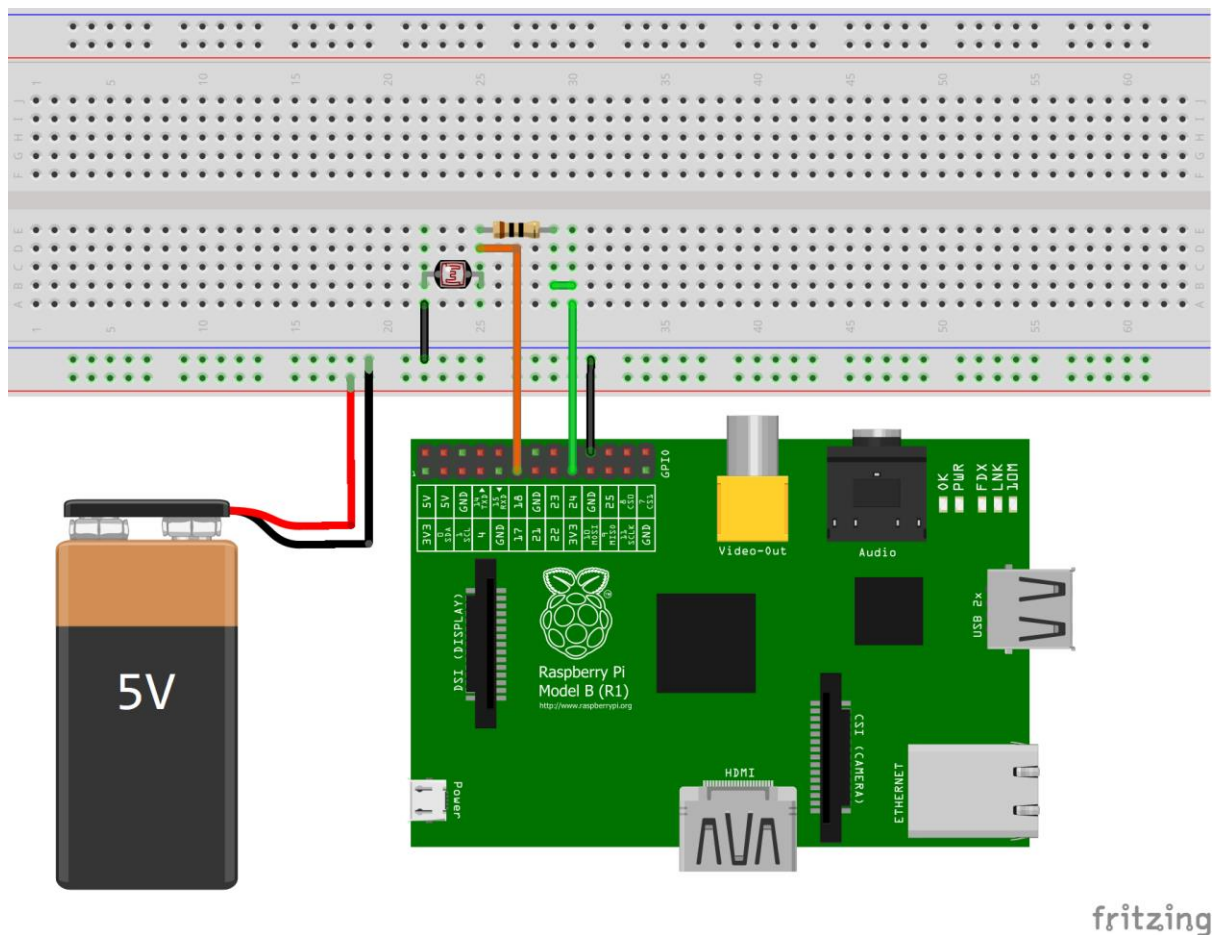


Fig. 2. Circuito Básico para el sensor LDR conectado a la Raspberry Pi.

3.1.2.Sensor digital (CNY70):

Como en el anterior circuito, queremos analizar los valores que se obtienen con el sensor CNY70 con el polímetro y con la lectura digital de la Raspberry Pi. Conectamos el sensor con las resistencias necesarias y la salida entre las resistencias 10K ohm y 15K ohm con el polímetro. Con este sensor medimos los datos que se obtienen con el polímetro al acercarle a cierta distancia un elemento blanco o negro.

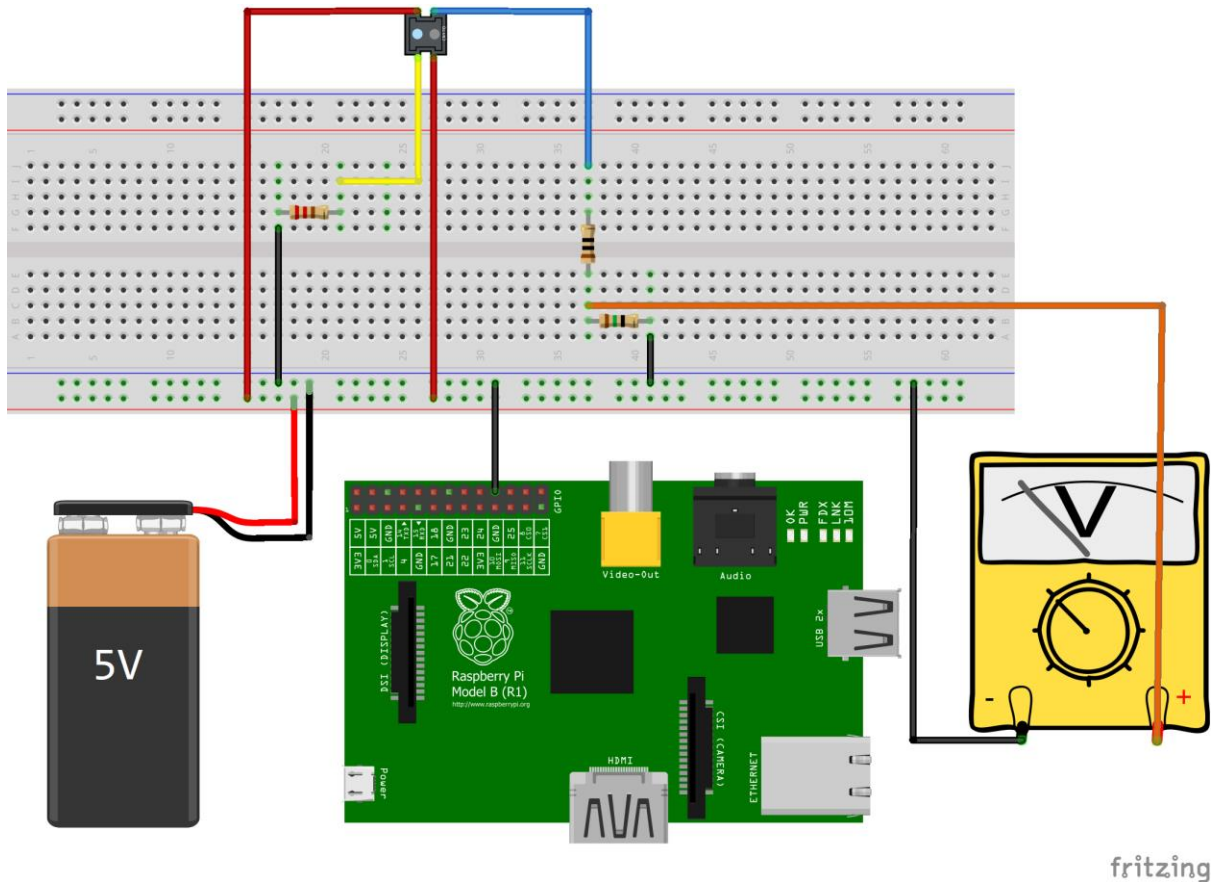


Fig. 3. Circuito Básico para el sensor CNY70 conectado al polímetro

Luego cambiamos un poco el circuito conectando la salida entre las resistencias 10K ohm y 15K ohm con una entrada del GPIO de la raspberry, en este caso elegimos el GPIO 17 (el Wiring Pin 0). Realizamos el mismo análisis anterior usando el mismo código que el anterior sensor.

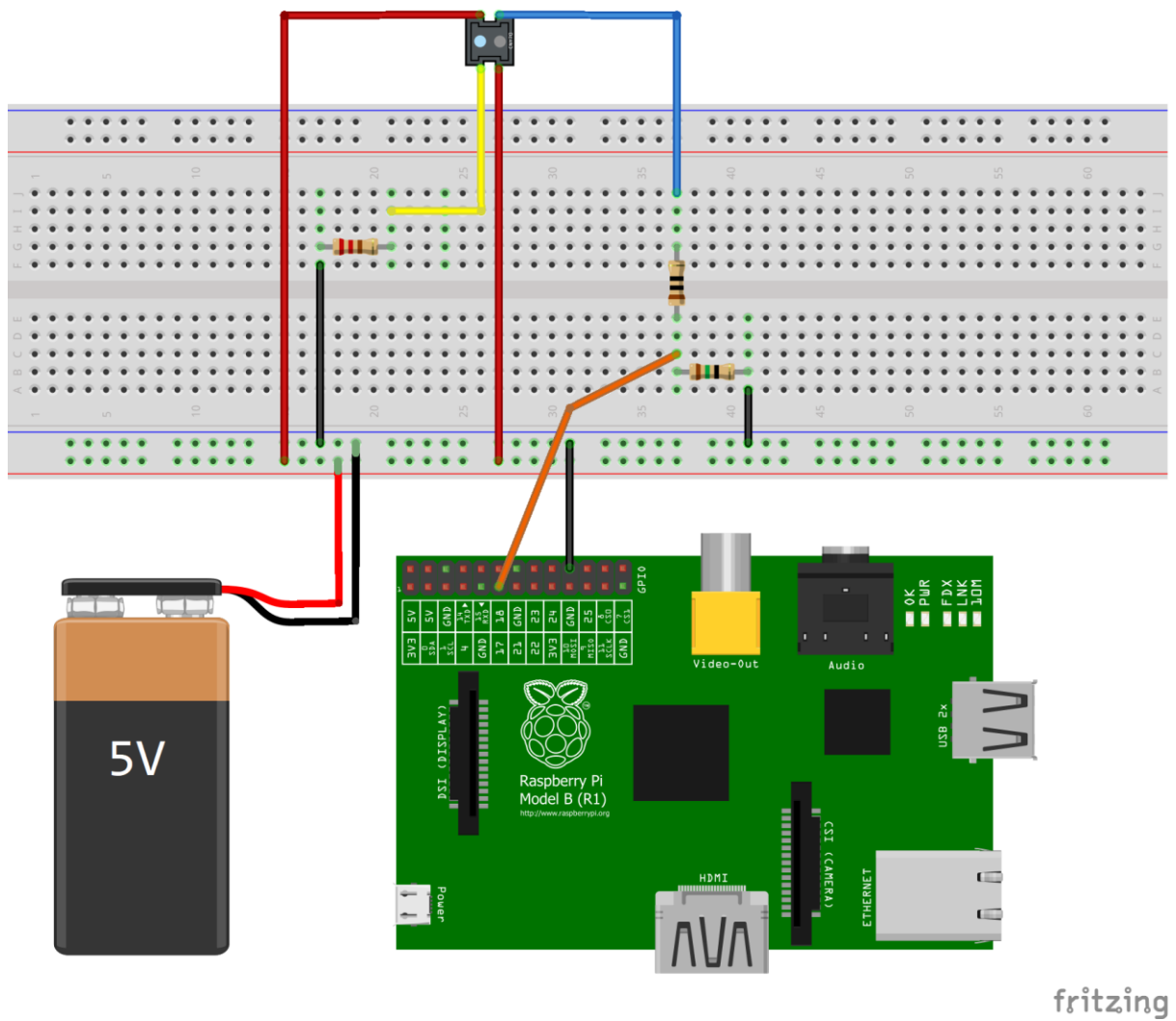
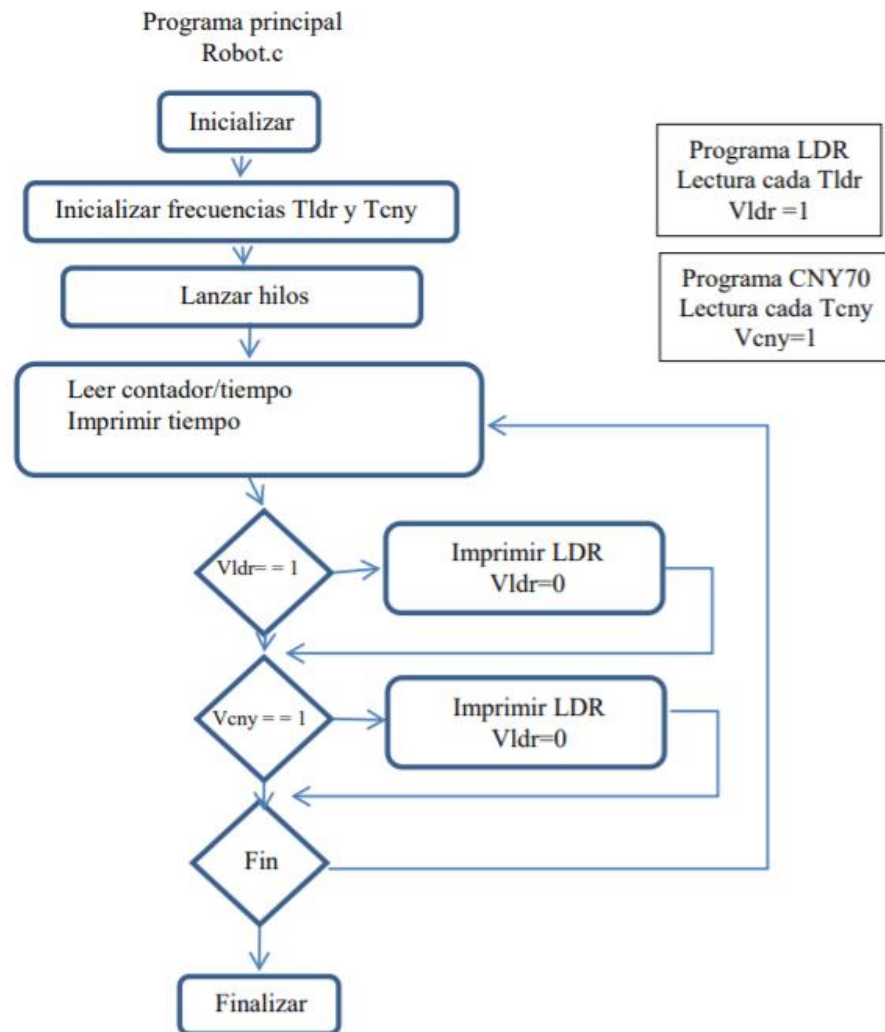


Fig. 4. Circuito Básico para el sensor CNY70 conectado a la RaspBerry Pi

3.2. Arquitectura software del robot

Finalmente, unimos los dos circuitos usados en el anterior apartado para realizar la segunda parte de la práctica. Para esta, se hizo un código siguiendo el siguiente diagrama:



Se usaron las siguientes funciones para generar hilos:

- pthreadCreate
- pi_Thread
- piLock
- piUnlock
- delay()

De esta manera el programa principal pueda detectar que hubo un cambio en los valores Vldr o Vcny que son los que indican que hay una lectura disponible.

Código:

```
• #include <wiringPi.h>
• #include <stdio.h>
• #include <stdlib.h>
• #include <softPwm.h>
• #include <pthread.h>
•
```

```

• #define PIN_LDR 0
• #define PIN_CNY 1
• int Vldr = 0;
• int Vcny = 0;
• int ldr = 0;
• int cny = 0;
•
•
•
•
• PI_THREAD(pin_ldr) {
•     int Tldr = 100;
•     int aLdr=0;
•     while(1) {
•         piLock(PIN_LDR);
•         aLdr=digitalRead(PIN_LDR);
•         if(ldr!=aLdr)
•         {ldr=aLdr;
•           Vldr=1;
•         }
•         piUnlock(PIN_LDR);
•         delay (Tldr);
•     }
•     return(0);
• }
•
• PI_THREAD(pin_cny) {
•     int Tcny = 150;
•     int aCny=0;
•     while(1) {
•         piLock(PIN_CNY);
•         aCny=digitalRead(PIN_CNY);
•         if(cny!=aCny)
•         {cny=aCny;
•           Vcny=1;
•         }
•         piUnlock(PIN_CNY);
•         delay (Tcny);
•     }
•     return(0);
• }
•
•
•
• int main(void){
•     wiringPiSetup();
•
•     pinMode (PIN_LDR, INPUT) ;
•     pinMode (PIN_CNY, INPUT) ;
•
•     int cont=0;
•     int tiempoFinal=1000;
•
•
•
•     int t = piThreadCreate(pin_ldr);
•     if (t != 0) {

```

```

•     printf("Thread did not start.\n");
•     }
•
•     t = pthreadCreate(&pin_cny);
•     if (t != 0) {
•         printf("Thread did not start.\n");
•     }
•
•     while(cont<tiempoFinal){
•
•         pthreadLock(PIN_CNY);
•
•         if(Vcny==1){
•             printf("Valor CNY: %d\n", cny);
•             Vcny=0;
•         }
•         pthreadUnlock(PIN_CNY);
•
•         pthreadLock(PIN_LDR);
•
•         if(Vldr==1){
•             printf("Valor LDR: %d\n", ldr);
•             Vldr=0;
•         }
•         pthreadUnlock(PIN_LDR);
•
•         delay(1000);
•         cont= cont+1;
•     }
•
•     return 0;
•     }

```

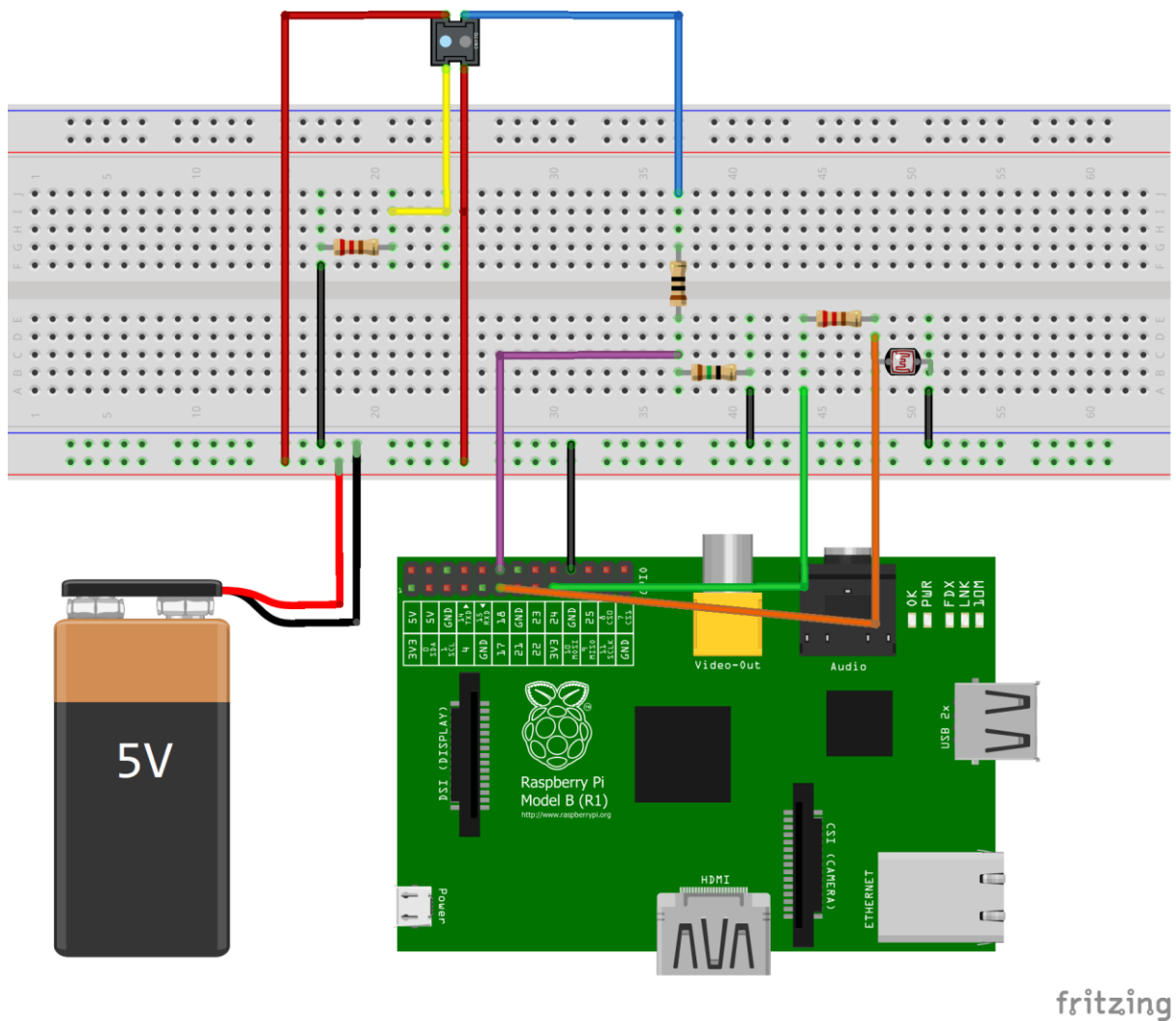


Fig. 5. Circuito Básico para el sensor LDR y CNY70 conectado a la RaspBerry Pi

4. Resultados y Análisis de Resultados

4.1. Lectura de datos analógicos con la RaspBerry Pi.

4.1.1. Resistencia variable con la luz (LDR)

4.1.1.1. Medir con el polímetro el valor de voltaje cuando:

- a) **Está iluminado únicamente con luz ambiente**
0.470 v
- b) **Iluminar con una fuente de luz el LDR (por ejemplo, con el led de un móvil)**
0.083 v
- c) **Se tapa el LDR para que esté a oscuras**
1.874 v

4.1.1.2. Medir con la RaspBerry Pi en los mismos casos anteriores. Indicar qué se obtiene. ¿Qué se puede concluir?

- a) **Está iluminado únicamente con luz ambiente**

Valor digital obtenido : 0

b) **Iluminar con una fuente de luz el LDR (por ejemplo con el led de un móvil)**

Valor digital obtenido :0

c) **Se tapa el LDR para que esté a oscuras**

Valor digital obtenido :1

4.1.2.Sensor digital (CNY70):

a) **Comprobar con el polímetro el rango de funcionamiento del sensor. Observe la distancia o alcance dependiendo del color del objeto; y los valores de voltaje que da cuando ve el color blanco y cuando ve el color negro.**

Utilizando una regla, una cartulina blanca y un plástico negro se comprobaron los valores que se obtienen al acercar elementos blancos y negros al sensor.

Primero iniciamos el experimento utilizando el polímetro midiendo los valores que se obtienen a cierta distancia:

COLOR BLANCO:

Distancia	Valor obtenido
0 cm	1.8 v
1 cm	1.4 v
2 cm	0.8 v
3 cm	0.6 v
5 cm	0.01 v

Nota: Estas pruebas se hicieron en una habitación suficientemente iluminada

COLOR NEGRO:

El máximo valor obtenido si acercamos el objeto al sensor, incluso pegado a este fue de 0.2 v, el valor en promedio era de 0.1 v

b) **Conecte la entrada digital de la RaspBerry Pi en el punto out y verifique que mide cuando se le muestra un objeto negro y uno blanco. Compruebe el efecto según a la distancia que se coloque el objeto.**

COLOR BLANCO:

Distancia	Valor obtenido
0 cm	1
1 cm	1
2 cm	0
3 cm	0

5 cm	0
------	---

Nota: Estas pruebas se hicieron en una habitación suficientemente iluminada

COLOR NEGRO:

Obtuvimos durante cualquier distancia el valor “0”.

Una cuestión importante a mencionar es que si se le acerca un elemento negro pero capaz de reflejar la luz emisora por el lector, el sensor podría hacer una lectura y obtener un valor de 1

4.2. Arquitectura software del robot

Luego de hacer el make del programa y ejecutarlo, se verificó que realizaba lo que le correspondía según la situación. Inicia con valores de LDR =0, es decir que recibe alguna luz el sensor LDR y CNY=0, no está detectando el rayo reflejado el sensor CNY70

- “Valor LDR: 1”: quiere decir que el LDR no está detectando luz
- “Valor LDR: 0”: quiere decir que el LDR está detectando luz
- “Valor CNY: 1”: quiere decir que el CNY70 está detectando un objeto de color claro (blanco) cercano
- “Valor CNY: 0”: quiere decir que el CNY70 no está detectando un objeto de color claro (blanco) cercano. Incluso puede tener un objeto negro cercano.

```
^Cpi@raspberrypi:~/wiringPi/examples $ sudo ./wfi
Valor LDR: 1
Valor CNY: 1
Valor CNY: 0
Valor LDR: 0
Valor CNY: 1
Valor LDR: 1
Valor CNY: 0
Valor LDR: 0
^Cpi@raspberrypi:~/wiringPi/examples $
```

Fig. 6. Ejecución del código del experimento II

5. Discusión y conclusiones

5.1. Conclusiones:

- Cuando el sensor LDR lee digitalmente un “1” y posee mayor voltaje es porque está en la oscuridad y cuando no lo está, lee 0 y posee un menor voltaje. Esto es así debido a que la resistencia del LDR es baja al recibir luz y muy alta al estar en la oscuridad.
- En el sensor CNY70, a mayor distancia con el color blanco, menor voltaje se lee. Cuando tiene cerca el color blanco, lee digitalmente un “1”, sino, lee “0”. Con el color negro lee siempre “0”. Esto es así ya que para poder detectar el rayo reflejado debe estar a una corta distancia y el objeto debe ser de un objeto claro para poder reflejar.

5.2. Errores:

No tuvimos errores en la realización de este experimento, pero reconocemos que pudo haber surgido varios problemas si no conectamos las tierras correctamente o si leíamos dónde no se debe los datos de los sensores.

6. Bibliografía

- PRÁCTICA: Programación de sensores en robótica. (2020). *Práctica – Sensores I*.