

**Parte 2**

# Lección

# 13

*Módulo de receptor IR*

## Resumen

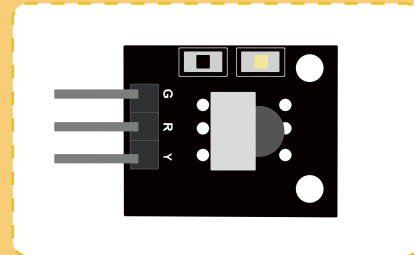
Usando un control remoto es una gran manera de tener el control inalámbrico de su proyecto.

Los mandos a distancia infrarrojos son simples y fáciles de usar. En este tutorial nos conectando el receptor IR para el UNO y luego use una biblioteca que fue diseñada para este sensor en particular.

En nuestro dibujo tenemos todos los códigos de IR Hexadecimal que están disponibles en este control remoto, también detectará si el código fue reconocido y también si estamos manteniendo pulsada una tecla

### Component Required:

- (1) x Elegoo Uno R3
- (1) x IR receiver module
- (1) x IR remote
- (3) x F-M cables (cables de hembra a macho DuPont)

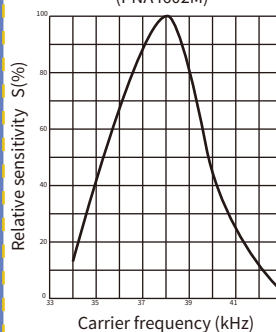


## Introducción del componente

### SENSOR RECEPTOR DE IR:

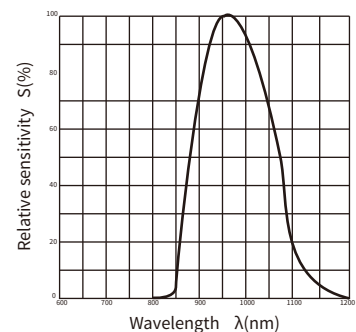
- Los** detectores infrarrojos son pequeños microchips con una célula fotoeléctrica que están configurados para recibir a la luz infrarroja. Casi siempre se utilizan para la detección de control remoto - cada TV y reproductor de DVD tiene uno de estos en la parte delantera para escuchar la señal de IR desde el clicker. Dentro del control remoto es un juego IR LED, que emite pulsos IR para comunicar al televisor para encender, apagar o cambiar de canal. Luz infrarroja no es visible para el ojo humano, lo que significa que tarda un poquito más de trabajo para probar una configuración
- Hay** algunas diferencias entre ellos y decir unas fotocélulas CdS: Detectores infrarrojos son especialmente filtrados para IR ligero, no son buenos para detectar luz visible. Por otro lado, las fotocélulas son buenas para detectar luz visible de amarillo/verde y no son buenas para luz IR.
- Detectores** infrarrojos tienen un demodulador en ese aspecto para IR modulada a 38 KHz. Sólo brilla un LED IR no detectada, tiene que ser PWM intermitente en 38 KHz. fotocélulas no tienen ningún tipo de demodulador y puede detectar cualquier frecuencia (incluyendo CC) dentro de la velocidad de respuesta de la fotocélula (que es aproximadamente 1KHz)
- Detectores** infrarrojos son salida digital - tampoco detectan señal 38KHz IR y salida bajo (0V) o no detecta ninguno y salida alto (5V). Fotocélulas actúan como resistencias, los cambios de resistencia dependiendo de cuánto se exponen a la luz
- Como** se puede ver en estos gráficos de hoja de datos, la detección de frecuencia de peak es a 38 KHz y el pico color del LED es de 940 nm. Se puede usar desde unos 35 KHz a 41 KHz pero la sensibilidad se desprenderá para que no detecte así desde lejos. Asimismo, puede utilizar LEDs de 850 a 1100 nm pero no funcionan tan bien como 900 a 1000nm así que asegúrese de obtener coincidencia de LEDs! Compruebe la ficha técnica para su IR LED verificar la longitud de onda.
- ¡Trate** de conseguir un 940nm - Recuerde que 940nm no es luz visible!

B.P.F frequency characteristics (PNA4602M)\*

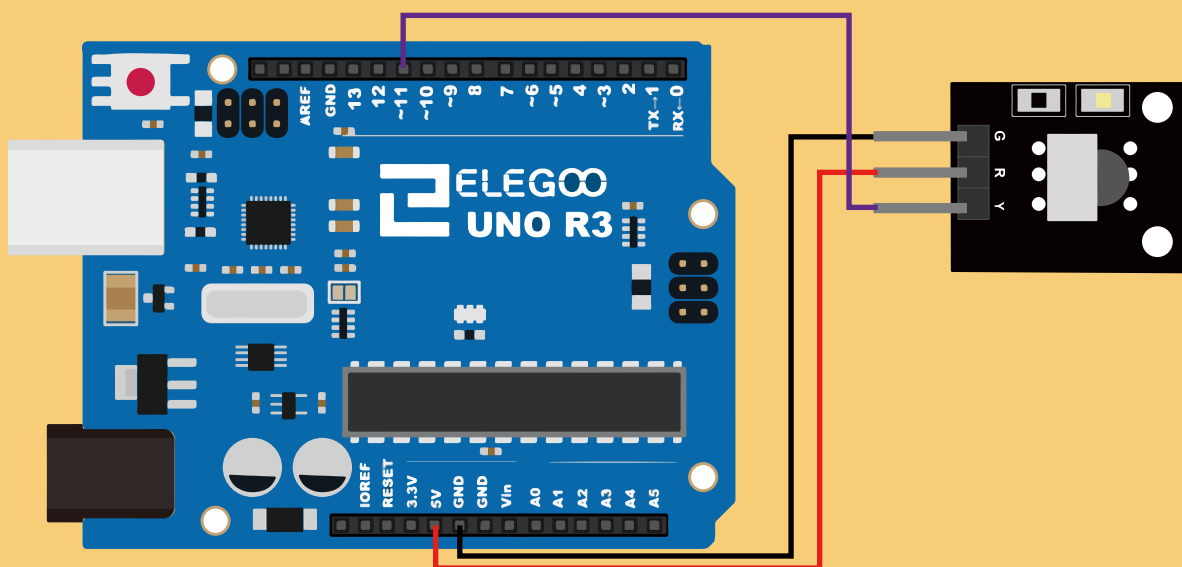
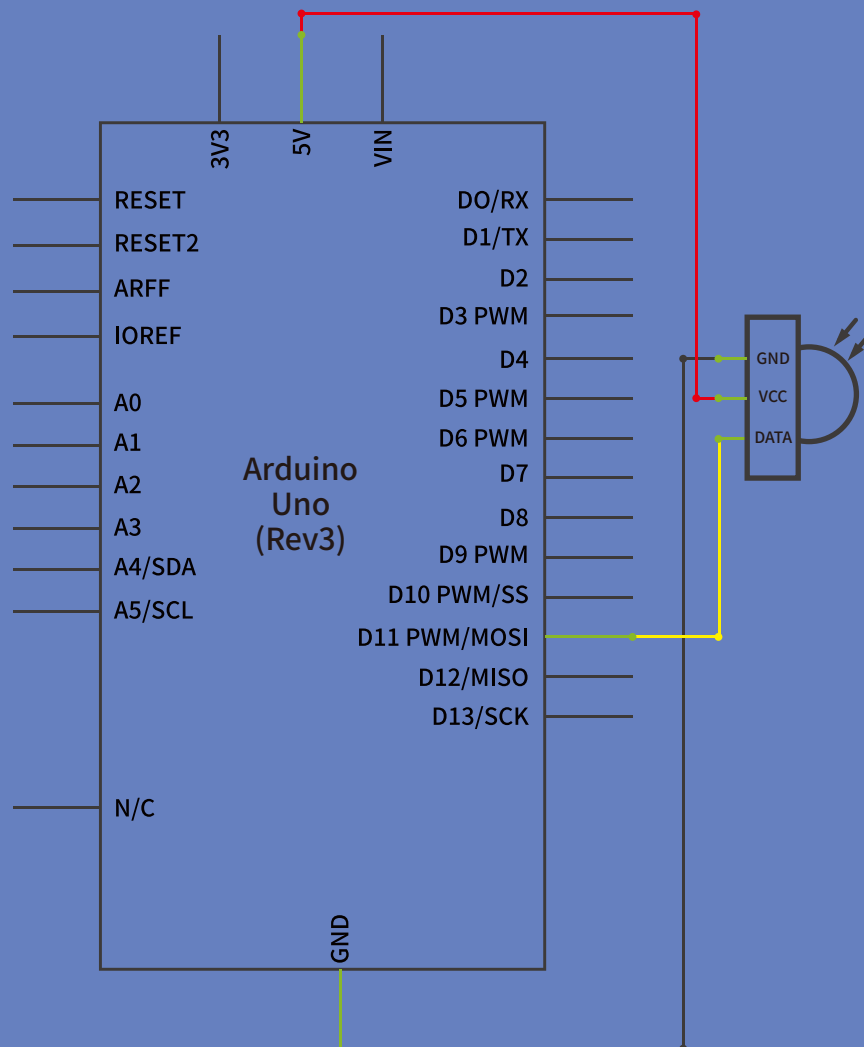


\* The peaks for PNA4601M, PNA4608M, and PNA4610M are all  $f_0$ .

Spectral sensitivity characteristics



## Conexión Esquema



Hay 3 conexiones para el receptor de infrarrojos.  
 Las conexiones son: señal, voltaje y tierra.  
 El "-" es la tierra, "S" es señal y medio es voltaje de 5V.

**Diagrama de cableado**

## Código

- Después del montaje, abra el programa en la carpeta “IR\_Receiver\_Module” donde se encuentra el curso y haga clic en CARGAR para cargar el programa. Consulte la Lección 5 en la parte 1 para obtener detalles sobre la carga del programa si hay algún error.
- A continuación nos trasladaremos el < RobotIRremote > fuera de la carpeta de biblioteca, para ello ya que entra en conflicto con la biblioteca va a utilizar. Sólo puede arrastrarlo dentro de la carpeta de la biblioteca una vez que haya terminado de programar el microcontrolador.
- Una vez que ha instalado la biblioteca, sólo seguir adelante y reiniciar su Software IDE.

switch...case

### [Estructura de control] Descripción

```
switch(results.value)

{
case 0xFFA25D: Serial.println("POWER"); break;
case 0xFFE21D: Serial.println("FUNC/STOP"); break;
.....
default:
Serial.println(" other button ");

} // End Case
```

- Al igual que las declaraciones “if”, “switch-case” controla el flujo de programas al permitir que los programadores especifiquen diferentes códigos que deben ejecutarse en diferentes condiciones. En particular, una declaración “switch” compara el valor de una variable con los valores especificados en la declaración “case”. Cuando se encuentra una declaración “case” cuyo valor coincide con el de la variable, se ejecuta el código que se especifique en esa declaración “case”.
- La palabra clave “break” sale de la instrucción “switch” y generalmente se usa al final de cada “case”. Sin una instrucción “break”, la instrucción “switch” continuará ejecutando las siguientes expresiones ("continuamente") hasta un corte, o hasta el final de la instrucción “switch”.

## Syntax

```
switch (var) {  
  case label1:  
    // declaraciones  
    break;  
  case label2:  
    // declaraciones  
    break;  
  default:  
    // declaraciones  
    break;  
}
```

## Parámetros

**var**: una variable cuyo valor es comparado en varios “case”. Tipos de datos permitidos: INT, CHAR.

**label1, label2**: constantes. Tipos de datos permitidos: INT, CHAR.

## Devuelve

Nada