

Detectores IR vs fotocélulas

Los detectores infrarrojos y las fotocélulas tienen diferentes características y usos específicos.

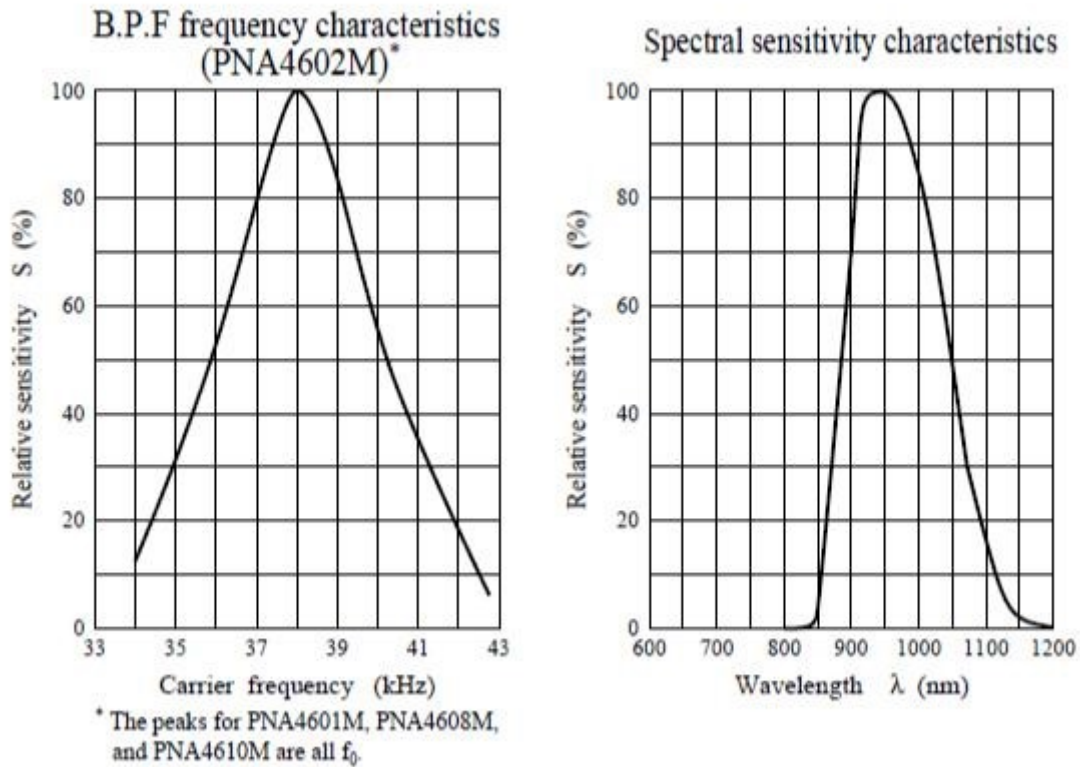
Los **detectores infrarrojos** están diseñados para detectar la luz infrarroja y están configurados para recibir señales moduladas en 38 KHz, como las señales de control remoto. Los detectores infrarrojos tienen una **salida digital** que indica si se detecta o no una señal de IR



Por otro lado, las **fotocélulas** pueden detectar luz visible en el espectro amarillo/verde y no están diseñadas específicamente para detectar luz infrarroja. Las fotocélulas actúan como resistencias y **cambian su resistencia** en función de la cantidad de luz a la que están expuestas.



¿Qué podemos medir?



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 kHz 41 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.

Esquema de conexiones

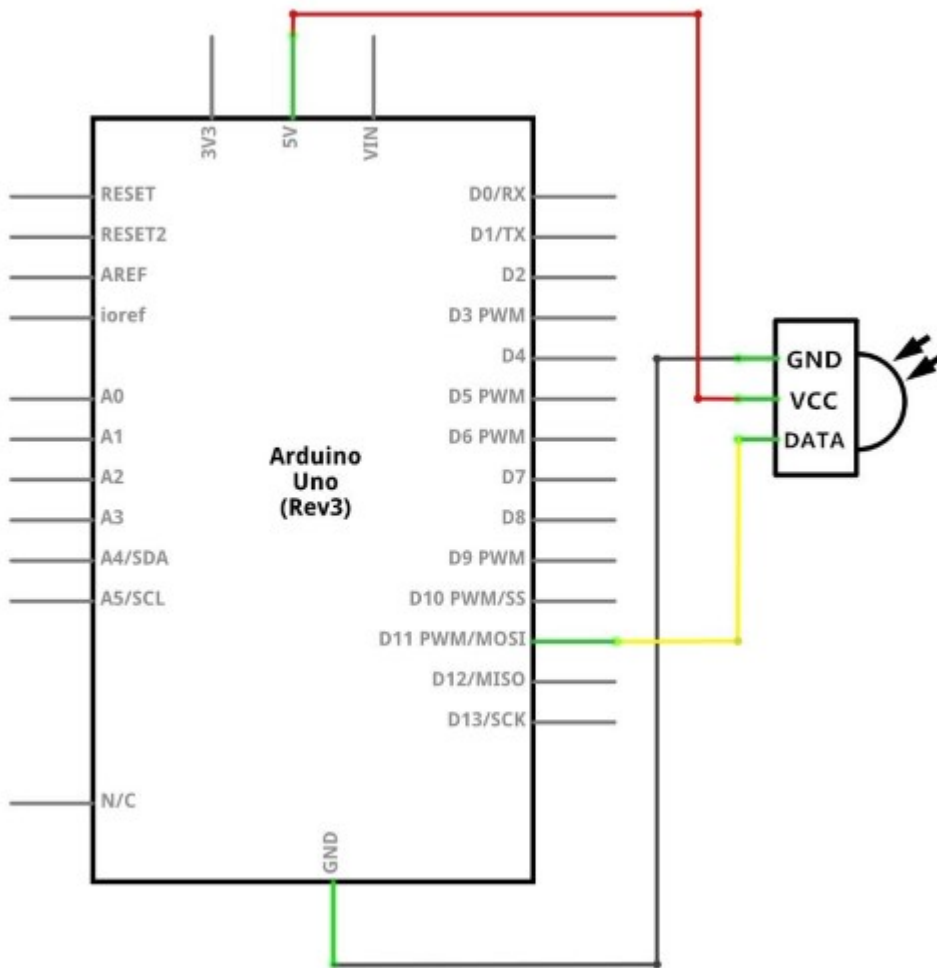
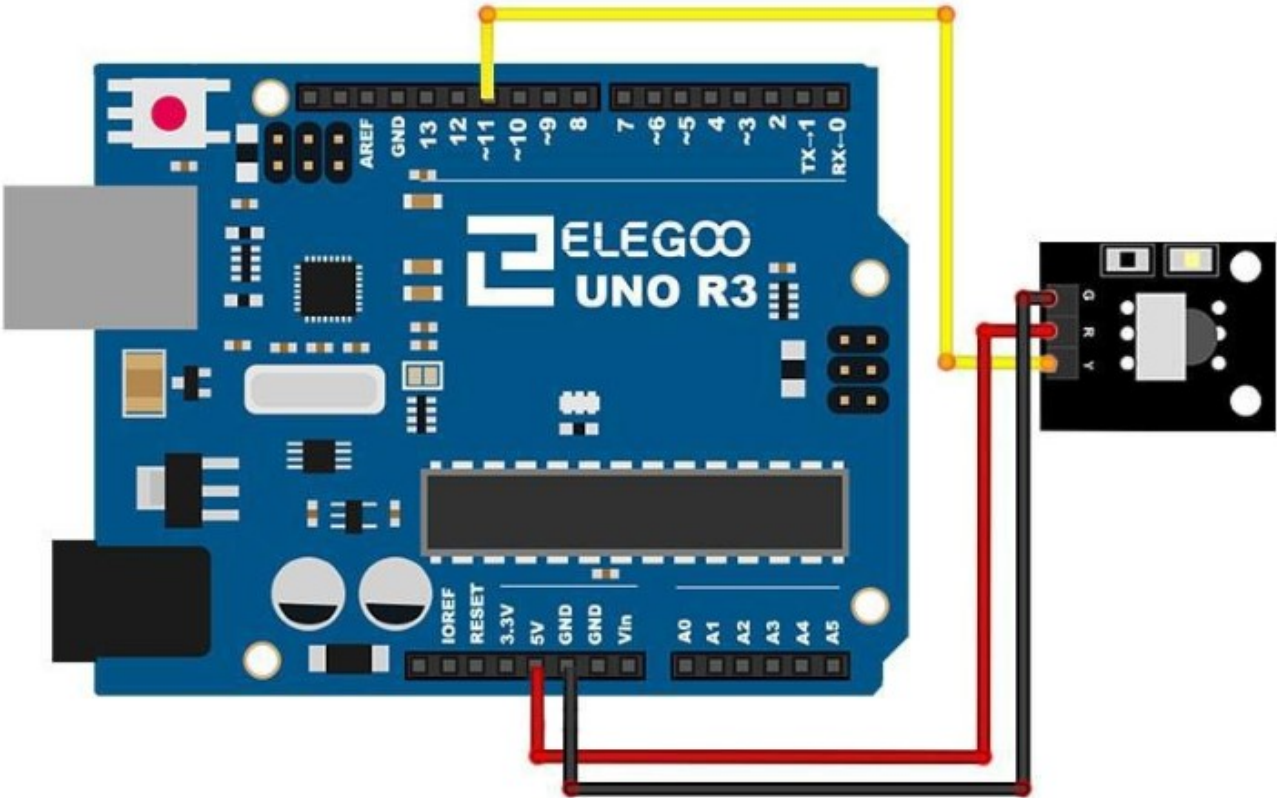


Diagrama de cableado

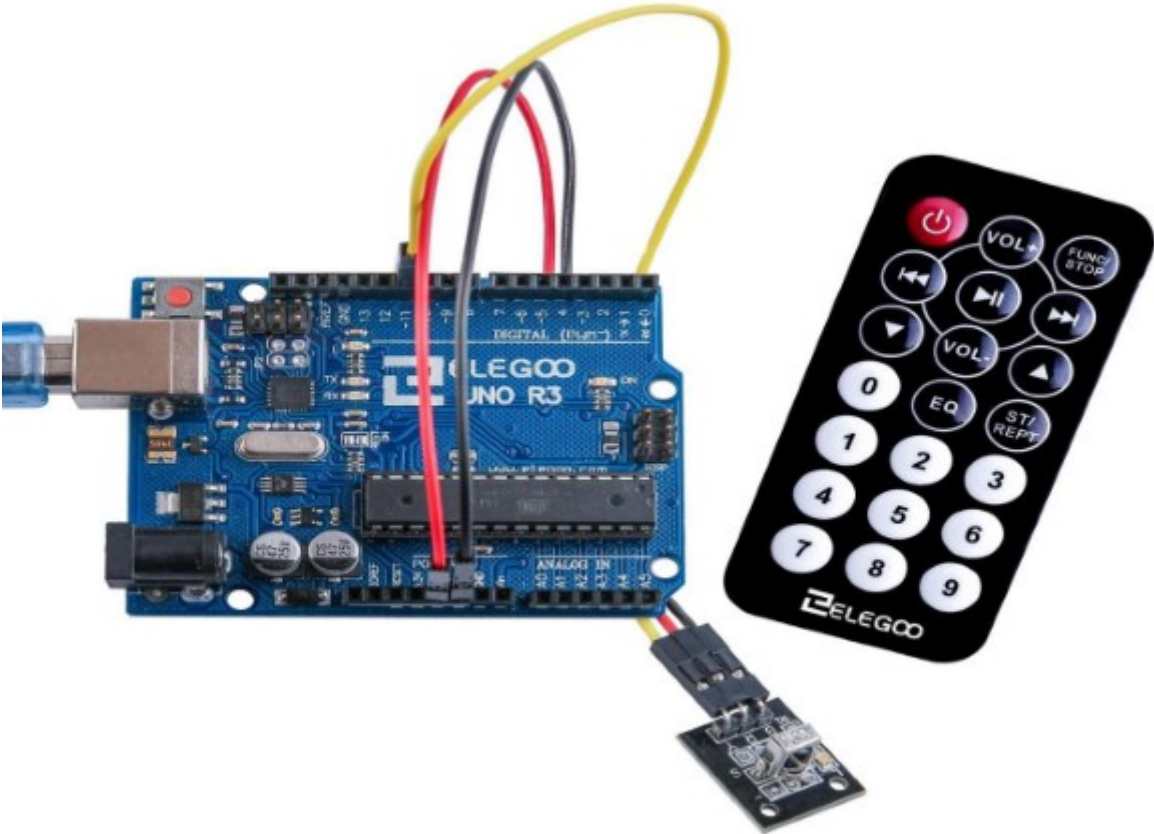
Hay 3 conexiones para el receptor de infrarrojos.

Las conexiones son: señal, voltaje y tierra.

- El "-" es la tierra
- "S" es señal
- El del medio corresponde a 5V.



Montaje



Código

```
#include "IRremote.h"

int receiver = 11; // Signal Pin of IR receiver

/*--( Declare objects )--*/
IRrecv irrecv(receiver); // create instance of 'irrecv'
decode_results results; // create instance of 'decode_results'

void translateIR()
{
    switch(results.value)
    {
        case 0xFFA25D: Serial.println("POWER"); break;
        case 0xFFE21D: Serial.println("FUNC/STOP"); break;
        case 0xFF629D: Serial.println("VOL+"); break;
        case 0xFF22DD: Serial.println("FAST BACK"); break;
        case 0xFF02FD: Serial.println("PAUSE"); break;
        case 0xFFC23D: Serial.println("FAST FORWARD"); break;
        case 0xFFE01F: Serial.println("DOWN"); break;
        case 0xFFA857: Serial.println("VOL-"); break;
        case 0xFF906F: Serial.println("UP"); break;
        case 0xFF9867: Serial.println("EQ"); break;
        case 0xFFB04F: Serial.println("ST/REPT"); break;
        case 0xFF6897: Serial.println("0"); break;
        case 0xFF30CF: Serial.println("1"); break;
        case 0xFF18E7: Serial.println("2"); break;
        case 0xFF7A85: Serial.println("3"); break;
        case 0xFF10EF: Serial.println("4"); break;
        case 0xFF38C7: Serial.println("5"); break;
        case 0xFF5AA5: Serial.println("6"); break;
        case 0xFF42BD: Serial.println("7"); break;
        case 0xFF4AB5: Serial.println("8"); break;
        case 0xFF52AD: Serial.println("9"); break;
        case 0xFFFFFFFF: Serial.println(" REPEAT");break;

        default:
            Serial.println(" other button ");
    }
    // End Case

    delay(500); // Do not get immediate repeat
} //END translateIR
void setup() /*--( SETUP: RUNS ONCE )--*/
{
    Serial.begin(9600);
    Serial.println("IR Receiver Button Decode");
    irrecv.enableIRIn(); // Start the receiver
}/*--(end setup )*/
```

```
void loop() /*-( LOOP: RUNS CONSTANTLY )-*/
{
  if (irrecv.decode(&results)) // have we received an IR signal?

  {
    translateIR();
    irrecv.resume(); // receive the next value
  }
}/* --(end main loop )-- */
```