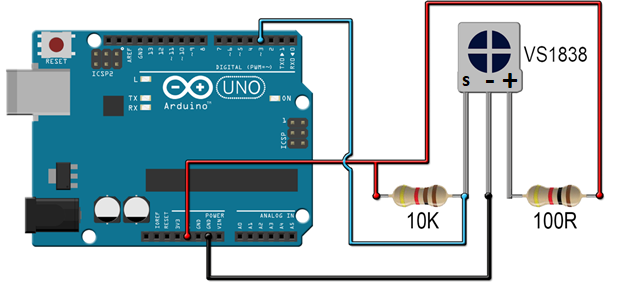
**IR Receiver with Arduino**

# PART 1 - Preparing the receiver.

We will use the VS1838 IR receiver. But, in order to work properly it needs a simple configuration that we could see below in the next schematic. We need to limit the current so a **100ohm** resistor is placed between Vcc pin and 5V. Also, the output needs a pull up so a 10K ohms resistor is placed between signal out pin and Vcc.  
  
PART 2 - Arduino schematic

Now the receiver will give a modulated signal each time it receives a IR light from the emitter. To read and decode the signal, we attach the VS1838 signal to digital pin D3 of an Arduino as seen below.



Now, the signal can be decoded with the Arduino. But the signal is strange and will be different for each type of remote. Below we have an example of the signal from the colors remote when the "ON" button is pressed. As you can see, the signal has different frequencies and we can't know which is a "1" and which a "0".  
  
For that, we will use a library for Arduino that will decode the signal and give us a numeric value for each button.

# PART 3 – Install IR Library in Arduino

1. Navigate to the [Releases](https://github.com/z3t0/Arduino-IRremote/releases) page.

2. Download the latest release.

3. Extract the zip file

4. Move the "IRremote" folder that has been extracted to your libraries directory.

5. Make sure to delete Arduino\_Root/libraries/RobotIRremote. Where Arduino\_Root refers to the install directory of Arduino. The library RobotIRremote has similar definitions to IRremote and causes errors.

# PART 4 – Testing IR Receiver with following code

#include <IRremote.h>

const int RECV\_PIN = 3;

IRrecv irrecv(RECV\_PIN);

decode\_results results;

void setup(){

Serial.begin(9600);

irrecv.enableIRIn();

irrecv.blink13(true);

}

void loop(){

if (irrecv.decode(&results)){

Serial.println(results.value, HEX);

switch (results.decode\_type){

case NEC: Serial.println("NEC"); break ;

case SONY: Serial.println("SONY"); break ;

case RC5: Serial.println("RC5"); break ;

case RC6: Serial.println("RC6"); break ;

case DISH: Serial.println("DISH"); break ;

case SHARP: Serial.println("SHARP"); break ;

case JVC: Serial.println("JVC"); break ;

case SANYO: Serial.println("SANYO"); break ;

case MITSUBISHI: Serial.println("MITSUBISHI"); break ;

case SAMSUNG: Serial.println("SAMSUNG"); break ;

case LG: Serial.println("LG"); break ;

case WHYNTER: Serial.println("WHYNTER"); break ;

case AIWA\_RC\_T501: Serial.println("AIWA\_RC\_T501"); break ;

case PANASONIC: Serial.println("PANASONIC"); break ;

case DENON: Serial.println("DENON"); break ;

default:

case UNKNOWN: Serial.println("UNKNOWN"); break ;

}

irrecv.resume();

}

}

Use serial monitor to see out put result.