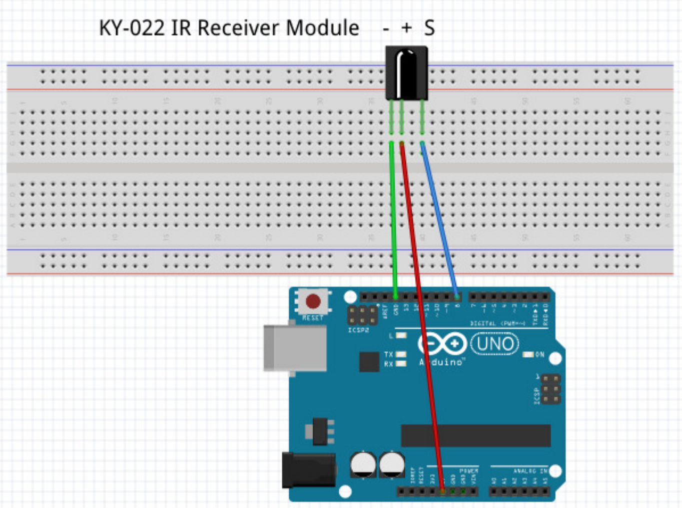
Reading an Infrared Remote Control

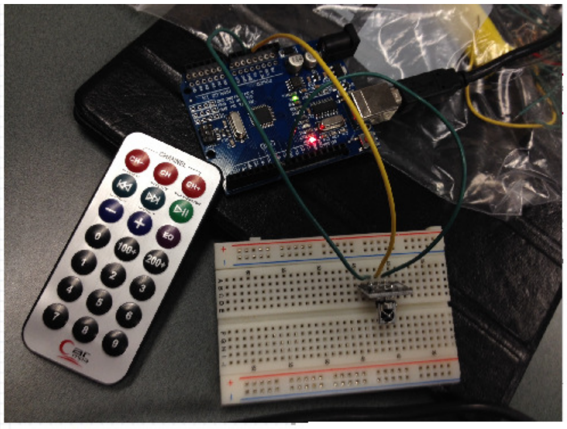
(G Payne – 2017)

You can use a simple infrared remote from a DVD player or TV remote to send codes to a KY-022 Infrared Remote Receiver. Then you can use the remote to control your Arduino application. Other infrared receivers will work. You will just need to know which pins to connect to which parts of the Arduino.

If you have not yet done so, you will need to go to GitHub and download the Arduino IRremote library and install it in your Arduino/libraries directory. NOTE: When you extract the .zip file, rename the folder to 'IRremote'.

Wire up the KY-022 to the Arduino as follows:





Connect Arduino pin 8 to the 'S' pin on the KY-022 receiver.
Connect Arduino GND to the '-' pin on the KY-022 receiver.
Connect Arduino +5V to the remaining pin on the KY-022 receiver.

NOTE: The code values that are defined in the following sketch are particular to the 'Car mp3' remote. Your particular remote may generate different codes. The sketch shows the raw codes received so you can adjust the sketch for the particular remote buttons you wish to use.

Depending on your particular remote, you may get '65535' or '255' codes when you press a button. It appears that those may be overflow conditions from holding the button down too long. The sketch has a small delay in it to slow things down a bit to try to minimize overflows. There are other ways you can do this should you not want to use the 'delay()' command.

Launch Arduino and start a fresh sketch. Name it ‘IRdemo’.

```
// demo for Infrared transmitter/receiver usage
// 2017 - Gordon Payne
// based on 'Car mp3' remote transmitter and KY- 022 Infrared Receiver Module
//
// KY pins  "-" to GND
//          "S" to pin 8
//          other pin to +5V

#include<IRremote.h>  //download and install this libray from gitHub,extract and
install the files in Arduino/libraries/IRremote
// MAKE SURE YOU DELETE the RemoteRobot library from your Arduino/libraries folder. It
contains similar commands

// the values pertaining to each key on the remote
// your remote will probably use different codes.
// variables for buttons 0 to 9
#define zero 26775
#define one 12495
#define two 6375
#define three 31365
#define four 4335
#define five 14535
#define six 23205
#define seven 17085
#define eight 19125
#define nine 21165
int receiver_pin = 8; // the ‘S’ pin on the KY-022 is connected to pin 8


IRrecv receiver(receiver_pin);// define a receiver object
decode_results output; // declare variable to store the output from the receiver

void setup()
{
  Serial.begin(9600); // open channel to the Serial Monitor
  receiver.enableIRIn(); // turn on the IR receiver

}

void loop() {
  if (receiver.decode(&output)) { // if a button code has been received from remote

    unsigned int value = output.value;
    Serial.println(value); // display the value for the remote button pressed
    switch (value) {
      case zero:
        Serial.println("0 pressed");
        break;
      case one:
        Serial.println("1 pressed");
        break;
      case two:
        Serial.println("2 pressed");
        break;
      case three:
        Serial.println("3 pressed");
        break;
      case four:
        Serial.println("4 pressed");
        break;
      case five:
        Serial.println("5 pressed");
        break;
      case six:
        Serial.println("6 pressed");
        break;
      case seven:
        Serial.println("7 pressed");
        break;
      case eight:
        Serial.println("8 pressed");
        break;
      case nine:
        Serial.println("9 pressed");
        break;
      default:
        Serial.println("undefined key or overflow");
        break;
    }
    delay(20); // slow down script a bit
    receiver.resume();// reset receiver for next code being sent from remote
  }
}
```

Now upload your sketch to the Arduino and open the Serial Monitor from the Tools menu.

If you aim the remote at the receiver and press any button 0 through 9, you should get messages indicating which button was pressed.

Now you can integrate remote control into your Arduino projects!

Now go out and MAKE SOMETHING AMAZING!!!!