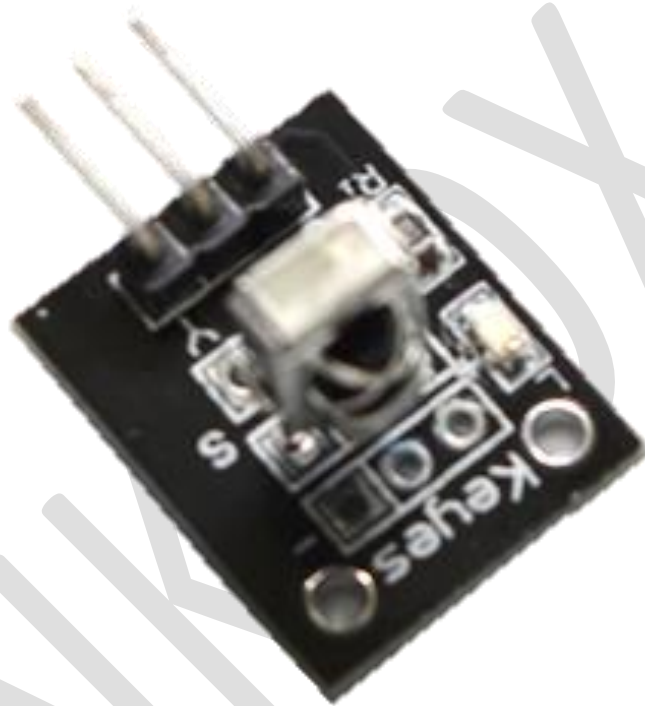


# Keyes

## Infrared Receiver Module



### General Description

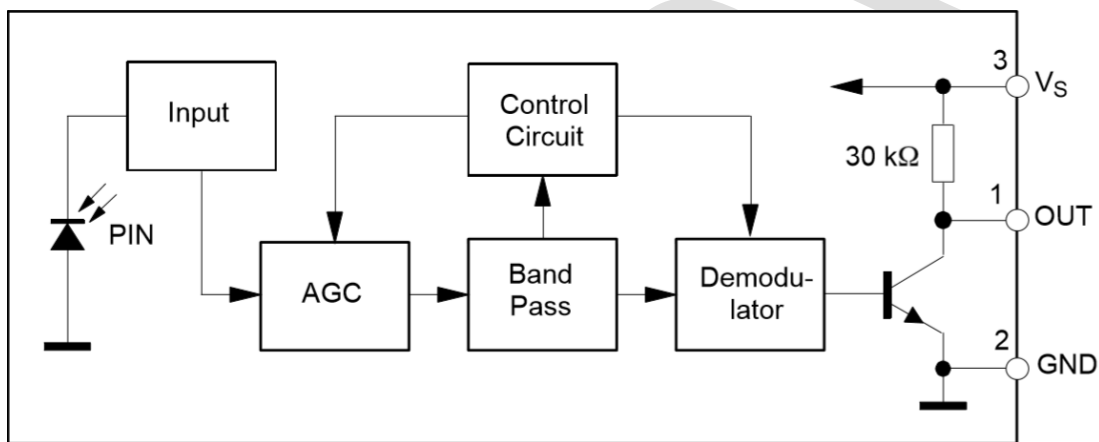
Use this simple Keyes IR receiver for infrared remote control of your next project. With low power consumption and an easy to use package, it mates well with embedded electronics and can be used with common IR remotes.

The TSOP18.. – series are miniaturized receivers for infrared remote control systems. PIN diode and preamplifier are assembled on lead frame, the epoxy package is designed as IR filter. The demodulated output signal can directly be decoded by a microprocessor. The main benefit is the reliable function even in disturbed ambient and the protection against uncontrolled output pulses.

## Specifications

- Supply Voltage: 2.7V to 5.5V
- Supply Current : 1.5mA
- Operating Temperature: -25°C to 85°C
- Frequency: 37.9KHZ
- Receiving Angle: 90°
- Receiving Distance: 18m

## Schematic

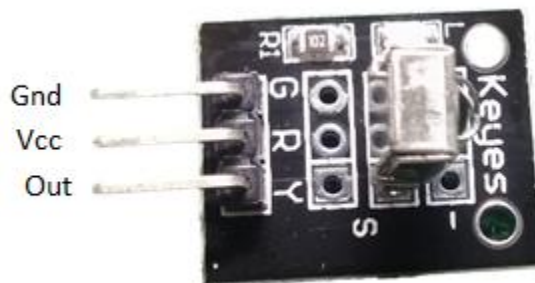


## Using the Module

You need:

- Arduino
- Keyes IR Receiver Module
- Connecting wires
- Any Remote / IR transmitter

1. Connect the Keyes IR Module to your arduino by following the pin connections shown below.



2. Download [IRremote library](#) and extract it to library folder in your Arduino directory.
3. Enter this sketch to your Arduino IDE then click upload. You can also find this at RFID library examples. This program will display the hex equivalent of the button pressed on a remote.

```
#include <IRremote.h>

int RECV_PIN = 11;

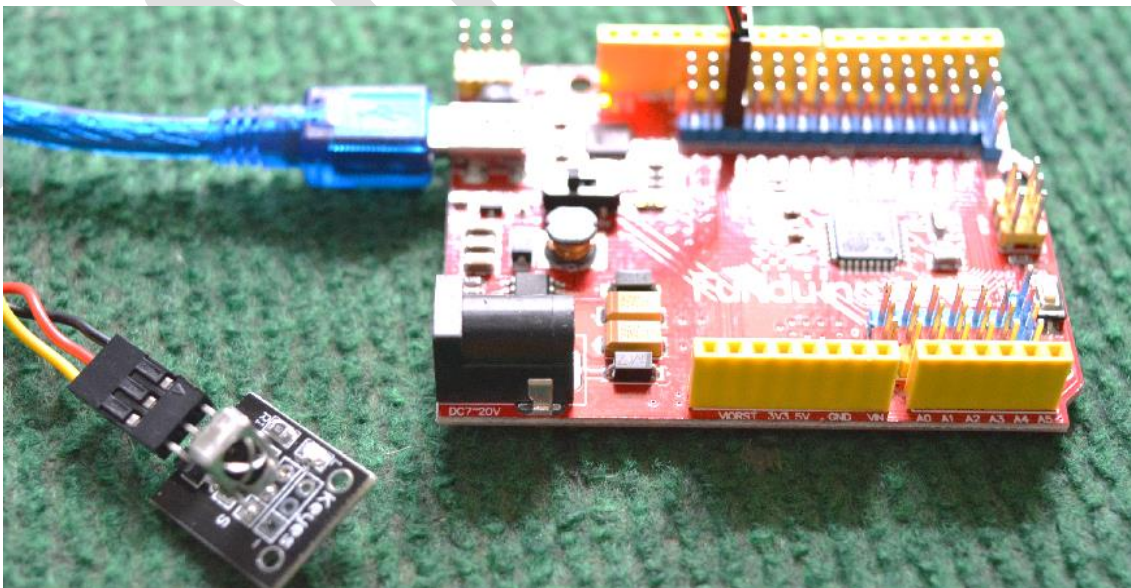
IRrecv irrecv(RECV_PIN);
decode_results results;

void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn(); // Start the receiver
}

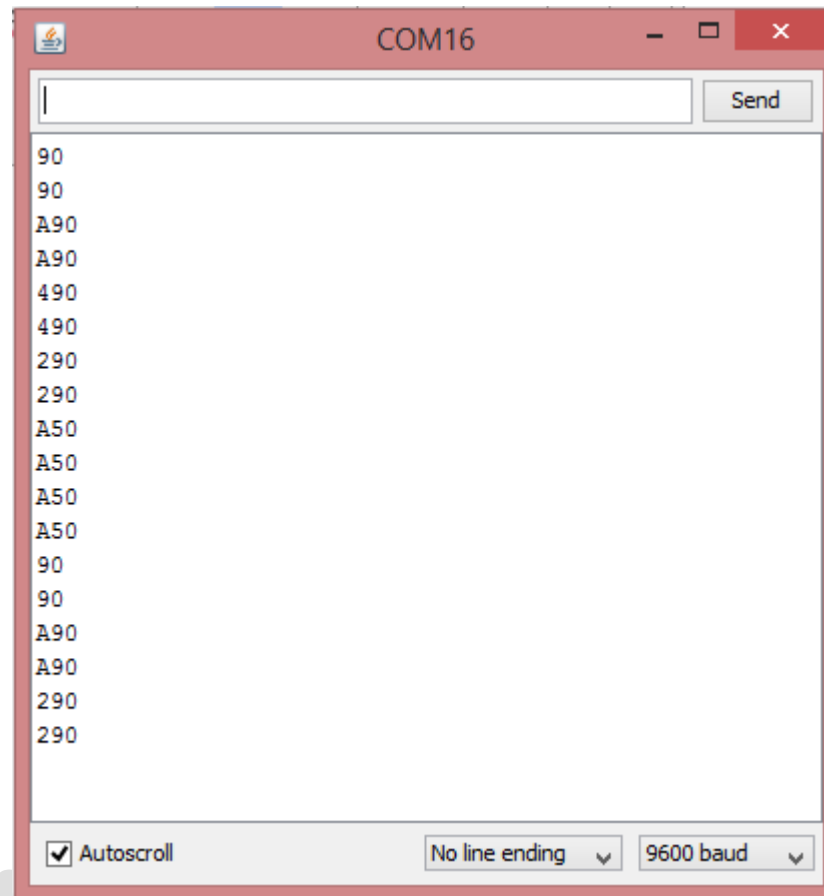
void loop() {
  if (irrecv.decode(&results)) {
    Serial.println(results.value, HEX);
    irrecv.resume(); // Receive the next value
    delay(200);
  }
}
```

4. Get your remote and press some buttons.

#### Actual Setup



## Results



## Sample Program

This sample program will assign a message on a remote button. I have used QRemote on android to transmit remote data.

```
#include <IRremote.h>

int message;
int RECV_PIN = 11;

IRrecv irrecv(RECV_PIN);

decode_results results;

void setup()
{
  Serial.begin(9600);
  irrecv.enableIRIn(); // Start the receiver
}
```

```

void loop() {
  if (irrecv.decode(&results)) {
    message = results.value, HEX;
    switch (message) {
      case 0xA90: Serial.println(" = Power ON"); break;
      case 0x290: Serial.println(" = Mute"); break;
      case 0x90:  Serial.println(" = Channel +"); break;
      case 0x890: Serial.println(" = Channel -"); break;
      case 0x490: Serial.println(" = Volume +"); break;
      case 0xC90: Serial.println(" = Volume -"); break;
      default:   Serial.println("Press a button");
    }
    irrecv.resume(); // Receive the next value
    delay(500);
  }
}

```

