



# Grove - Infrared Receiver User Manual

Release date: 2015/9/22

Version: 1.0

Wiki: [http://www.seeedstudio.com/wiki/Grove\\_-\\_Infrared\\_Receiver](http://www.seeedstudio.com/wiki/Grove_-_Infrared_Receiver)

Bazaar: [http://www.seeedstudio.com/depot/Grove-Infrared-Receiver-p-994.html?cPath=19\\_23](http://www.seeedstudio.com/depot/Grove-Infrared-Receiver-p-994.html?cPath=19_23)

## Document Revision History

---

Revision	Date	Author	Description
1.0	Sep 22, 2015	Loovee	Create file

## Contents

Document Revision History .....	2
1. Introduction .....	2
2. Specification .....	3
3. Demonstration .....	4
4. With Arduino/Seeeduino .....	5
4.1 Suggest Reading for Starter .....	5
4.2 IRSendRev Library .....	5
4.2.1 Setup .....	5
4.3 Infrared Receiver Examples/Applications .....	5
4.3.1 Receiver .....	5
4.3.2 Emitter .....	8
5. Resources .....	10

### *Disclaimer*

*For physical injuries and possessions loss caused by those reasons which are not related to product quality, such as operating without following manual guide, natural disasters or force majeure, we take no responsibility for that.*

*Under the supervision of Seeed Technology Inc., this manual has been compiled and published which covered the latest product description and specification. The content of this manual is subject to change without notice.*

### *Copyright*

*The design of this product (including software) and its accessories is under tutelage of laws. Any action to violate relevant right of our product will be penalized through law. Please consciously observe relevant local laws in the use of this product.*

## 1. Introduction

---

The Infrared Receiver is used to receive infrared signals and also used for remote control detection. There is an IR detector on the Infrared Receiver which is used to get the infrared light emitted by the Infrared Emitter. The IR detector have a demodulator inside that looks for modulated IR at 38 KHz. The Infrared Receiver can receive signals well within 10 meters. If more than 10 meters, the receiver may not get the signals. We often use the two Groves-the Infrared Receiver and the [Grove - Infrared Emitter](#) to work together.



## 2. Specification

---

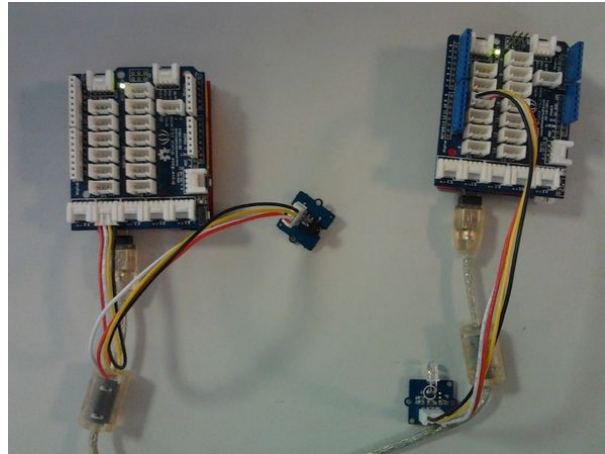
- Voltage: 3.3-5V
- Distance: 10m

### 3. Demonstration

---

We will use Grove-Infrared Receiver and Grove - Infrared Emitter in this demonstration. The Infrared Receiver will receive the data that the Grove - Infrared Emitter sends.

- Connect the Grove - Infrared Emitter to D3;
- Connect the Grove - Infrared Receiver to D2;



## 4. With Arduino/Seeeduino

---

### 4.1 Suggest Reading for Starter

- [Download Arduino and install Arduino driver](#)
- [Getting Started with Seeeduino/Arduino](#)

### 4.2 IRSendRev Library

We have created a library to help you start playing quickly with the Seeeduino/Arduino, in this section we'll show you how to set up the library.

#### 4.2.1 Setup

- [\*Download the library code as a zip file from the IRSendRev github page.\*](#)
- Unzip the downloaded file into your `.../arduino/libraries`.
- Rename the unzipped folder "IRSendRev"
- Start the Arduino IDE (or restart if it is open).

### 4.3 Infrared Receiver Examples/Applications

These example are going to show you how to use features of Grove - Infrared Receiver. You can use Infrared Receiver combination with [Infrared Emitter Grove](#). Connect the IR receiver pins to D2 for this demo.

#### 4.3.1 Receiver

- Open File->Examples->IRSendRev->example->recv sketch for a complete example, or copy and paste code below to a new Arduino sketch.

##### *Description:*

This example connect the IR receiver pins to D2 for this demo. You can see the remote control's infrared data that received through a serial port terminal, then write the received infrared data into send.ino and upload to the board with Infrared Emitter Grove, so you can send the same data with remote control's button.

##### *Application:*

You can note the remote control's infrared data down through Infrared Receiver, then send the



same data through Infrared Emitter in some cases, such as open the fan switch when indoor temperature is greater than 26 degrees.

```
#include <IRSendRev.h>

#define BIT_LEN      0
#define BIT_START_H  1
#define BIT_START_L  2
#define BIT_DATA_H    3
#define BIT_DATA_L    4
#define BIT_DATA_LEN  5
#define BIT_DATA      6

const int pinRecv = 2;          // ir receiver connect to D2

void setup()
{
    Serial.begin(115200);
    IR.Init(pinRecv);
    Serial.println("init over");
}

unsigned char dta[20];

void loop()
{
    if(IR.IsDta())                // get IR data
    {
        IR.Recv(dta);            // receive data to dta

        Serial.println("+-----+");
        Serial.print("LEN = ");
        Serial.println(dta[BIT_LEN]);
        Serial.print("START_H: ");
        Serial.print(dta[BIT_START_H]);
        Serial.print("\tSTART_L: ");
        Serial.println(dta[BIT_START_L]);

        Serial.print("DATA_H: ");
        Serial.print(dta[BIT_DATA_H]);
        Serial.print("\tDATA_L: ");
        Serial.println(dta[BIT_DATA_L]);

        Serial.print("\r\nDATA_LEN = ");
        Serial.println(dta[BIT_DATA_LEN]);
    }
}
```

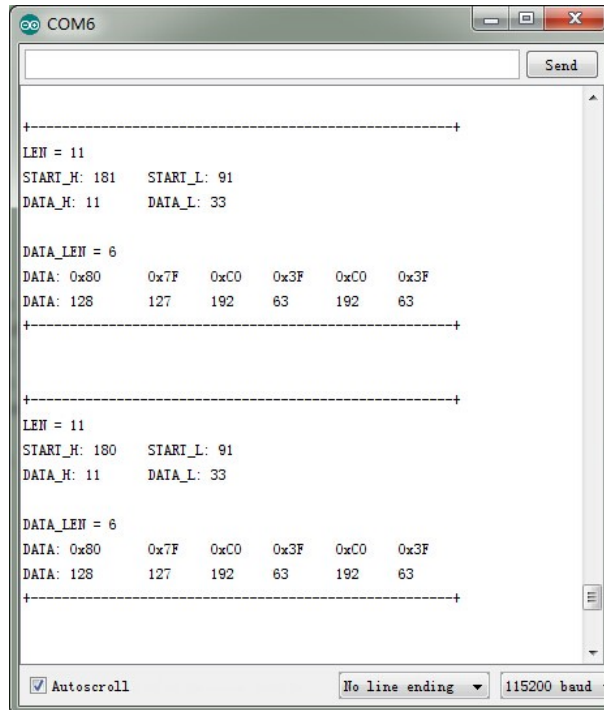
```

Serial.print("DATA: ");
for(int i=0; i<dta[BIT_DATA_LEN]; i++)
{
    Serial.print("0x");
    Serial.print(dta[i+BIT_DATA], HEX);
    Serial.print("\t");
}
Serial.println();

Serial.print("DATA: ");
for(int i=0; i<dta[BIT_DATA_LEN]; i++)
{
    Serial.print(dta[i+BIT_DATA], DEC);
    Serial.print("\t");
}
Serial.println();
Serial.println("+-----+\r\n\r\n");
}
}

```

- Upload the code to the development board.
- Open the serial monitor window and wait for the input.
- Using IR remote control sending data ([This example use MIDEA Company's IR remote control of fans, and press the open/close key.](#)).
- You can see the information below.



### 4.3.2 Emitter

Open **File->Examples->IRSendRev->example->send** sketch for a complete example, or copy and paste code below to a new Arduino sketch.

#### Description:

Connect the IR send pins to D3 for this demo. You can see the remote control's infrared data that received through Infrared Receiver, such as the example above. Then write the received infrared data into this example and upload to the board with [Infrared Emitter Grove](#), so you can send the same data with remote control's button.

#### Application

You can note the remote control's infrared data down through Infrared Receiver, then send the same data through Infrared Emitter in some cases, such as open the fan switch when indoor temperature is greater than 26 degrees.

**Notice:** Must connect the IR send pins to D3 for this demo.

```
#include <IRSendRev.h>

#define BIT_LEN      0
#define BIT_START_H  1
#define BIT_START_L  2
#define BIT_DATA_H    3
```

```

#define BIT_DATA_L      4
#define BIT_DATA_LEN    5
#define BIT_DATA        6

const int ir_freq = 38;          // 38k

unsigned char dtaSend[20];

void dtaInit()
{
    dtaSend[BIT_LEN]      = 11;          // all data that needs to be sent
    dtaSend[BIT_START_H]  = 180;         // the logic high duration of "Start"
    dtaSend[BIT_START_L]  = 91;          // the logic low duration of "Start"
    dtaSend[BIT_DATA_H]   = 11;          // the logic "long" duration in the communication
    dtaSend[BIT_DATA_L]   = 33;          // the logic "short" duration in the communication

    dtaSend[BIT_DATA_LEN] = 6;           // Number of data which will sent. If the number is
other, you should increase or reduce dtaSend[BIT_DATA+x].

    dtaSend[BIT_DATA+0]   = 128;         // data that will sent
    dtaSend[BIT_DATA+1]   = 127;
    dtaSend[BIT_DATA+2]   = 192;
    dtaSend[BIT_DATA+3]   = 63;
    dtaSend[BIT_DATA+4]   = 192;
    dtaSend[BIT_DATA+5]   = 63;
}

void setup()
{
    dtaInit();
}

void loop()
{
    IR.Send(dtaSend, 38);

    delay(2000);
}

```

## 5. Resources

---

- [Grove - Infrared Receiver eagle files](#)
- [IR Send and Receiver Library](#)
- [IR Receive Library for LinkIt ONE](#)
- [TSOP282 Datasheet](#)



Компания «ЭлектроПласт» предлагает заключение долгосрочных отношений при поставках импортных электронных компонентов на взаимовыгодных условиях!

Наши преимущества:

- Оперативные поставки широкого спектра электронных компонентов отечественного и импортного производства напрямую от производителей и с крупнейших мировых складов;
- Поставка более 17-ти миллионов наименований электронных компонентов;
- Поставка сложных, дефицитных, либо снятых с производства позиций;
- Оперативные сроки поставки под заказ (от 5 рабочих дней);
- Экспресс доставка в любую точку России;
- Техническая поддержка проекта, помощь в подборе аналогов, поставка прототипов;
- Система менеджмента качества сертифицирована по Международному стандарту ISO 9001;
- Лицензия ФСБ на осуществление работ с использованием сведений, составляющих государственную тайну;
- Поставка специализированных компонентов (Xilinx, Altera, Analog Devices, Intersil, Interpoint, Microsemi, Aeroflex, Peregrine, Syfer, Eurofarad, Texas Instrument, Miteq, Cobham, E2V, MA-COM, Hittite, Mini-Circuits, General Dynamics и др.);

Помимо этого, одним из направлений компании «ЭлектроПласт» является направление «Источники питания». Мы предлагаем Вам помощь Конструкторского отдела:

- Подбор оптимального решения, техническое обоснование при выборе компонента;
- Подбор аналогов;
- Консультации по применению компонента;
- Поставка образцов и прототипов;
- Техническая поддержка проекта;
- Защита от снятия компонента с производства.



#### Как с нами связаться

**Телефон:** 8 (812) 309 58 32 (многоканальный)

**Факс:** 8 (812) 320-02-42

**Электронная почта:** [org@eplast1.ru](mailto:org@eplast1.ru)

**Адрес:** 198099, г. Санкт-Петербург, ул. Калинина, дом 2, корпус 4, литера А.