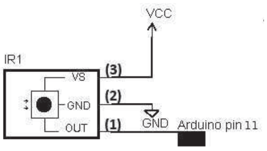
LEARN ! DO ! ArduinoInfo.Info

IR-RemoteControl (/IR-RemoteControl)   Editar   4 (/IR-RemoteControl#discussion)   62 (/page/history/IR-RemoteControl)   ... (/page/menu/IR-RemoteControl)

Infrared (IR) Remote

Here's the pinout for almost every 3-pin IR Receiver:

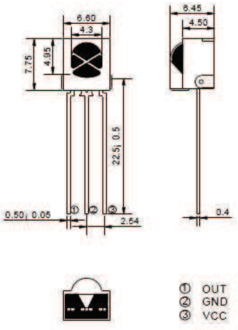


(Here) is a link to a typical IR Receiver Spec Sheet:

IR-Receiver-AX-1838HS.pdf
Details Download 380 KB

And here's another said to be more sensitive:

IR-tsop48.pdf
Details Download 278 KB



(Above): a diagram of connecting the receiver to an Arduino. You can get these HERE.

There are many different manufacturers of IR Receivers and some have different pinouts:

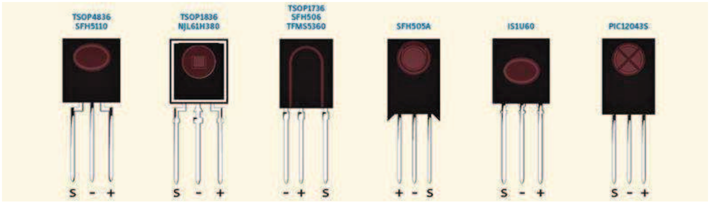


Image courtesy of Alberto Piganti. See: http://www.pighixxx.com/

There is also an easy-to-connect IR Receiver Electronic Brick like this (right). It can be plugged into a Sensor Shield or YourDuinoRobo1 with a 3-pin cable.

CONNECTION NOTE: The IR Remote Receiver Electronic Brick has 3 pins. From left to right they are: (G) Ground, (V) Voltage, (S) Signal. BUT the marking sometimes vary on the little circuit board. In this photo they are marked G-R-Y. The 3-pin cable in the photo has the typical color code: (G) Ground = Black, (V) Voltage = Red, (S) Signal = White.
This brick also comes with the IR Infrared Robot Remote Control Kit which has a remote with arrow buttons for direction etc. (Scroll down for example). It is also in the YourDuino Electronic Brick Set.

DETAILED IR REMOTE CONTROL INFORMATION (THANKS! to Sam Bergmans)

IR-REMOTE LIBRARY:

Note: The following library **must** be installed in your Arduino installation for this to work!
CLICK HERE - IR REMOTE CONTROL: ARDUINO LIBRARY

NOTE!! If you have a late version of Arduino with a library IRRobotRemote, it may conflict and you may have to remove that library. 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.

NOTE: For Info on easier Library Installs, SEE THIS:

More IR examples and projects on the IRemote wiki HERE:

TYPES OF IR REMOTE CONTROLS

NOTE!! Most handheld remotes are shipped with a small clear plastic piece in the battery compartment that must be removed to activate it. You can usually just pull it out.

There are many different IR remote controls. Some from YourDuino.com are the low-cost IR Infrared Remote Control Kit 2 and also the THIS IR Remote (right) which has directional buttons that would be good for controlling a vehicle etc. Then, there are the typical TV and Stereo Remotes. All of these may have different encoding methods and number of physical buttons, and different codes received when a button is pressed. Below we will give example Software Sketches for a few common IR Remotes.

IRrecvDemo SKETCH:Read codes from almost any IR Remote

If you need to discover the codes received from an unknown IR Remote type, use this Sketch from the IR Remote Control Library Examples first. (You **must** first install that library - the link is above).

(Copy and paste into a blank Arduino IDE Window), Upload to your Arduino and start the Serial Monitor window:

```
/*
 * IRremote: IRrecvDemo - demonstrates receiving IR codes with IRrecv
 * An IR detector/demodulator must be connected to the input RECV_PIN.
```

Búsqueda

HOME

ABOUT US!

YourDuinoShop

NEWSLETTERS

BLOG

THIS WIKI:

WHAT IS Arduino?

LEARNING Arduino:

- Arduino: WhatIsIt?
- Hands-On Learning
- Example Sketches
- Arduino Libraries

ARDUINO HOWTO:

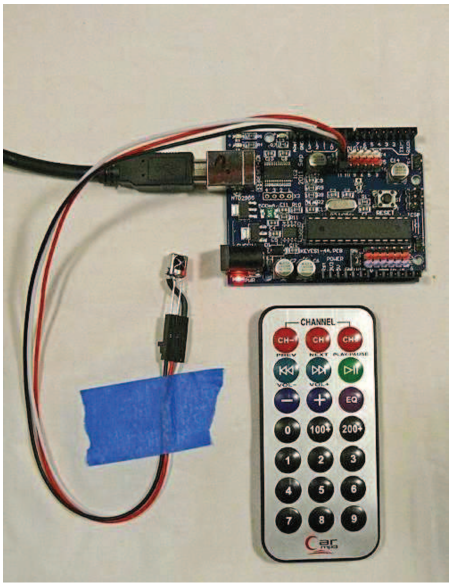
- CABLES, WIRES and PINS
- SENSOR SHIELD
- ARDUINO POWER
- SERVO MOTORS
- WIRELESS nRF24L01
- LCD DISPLAYS
- SD CARDS
- Ultrasonic Sensors
- TemperatureSensc
- IR Remotes
- Stepper Motors
- RealTimeClock
- Keyboards/Button
- PWM Frequencies
- Analog Output
- TUTORIALS

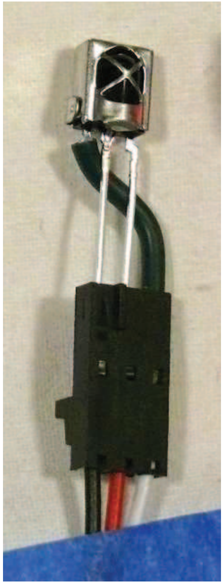
BJK
C

```
 * Version 0.1 July, 2009
 * Copyright 2009 Ken Shirriff
 * http://arcfn.com
 */

#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(100);
}
```

EXAMPLE: The YourDuino.com IR Infrared Remote Control Kit 2

Below is the IR Remote Control Kit connected to a YourDuinoRobo1 with a 3-pin cable. On the right is the detail of the way the IR Receiver is carefully plugged into Gnd and Vcc on the cable, and the Out pin is insulated with a piece stripped from another wire, the pins are cut off evenly, and Out is routed into the Signal (White) pin of the cable. The software below displays the button that was pressed.





Test Arduino Software Sketch for IR Infrared Remote Control Kit 2 (TESTED!!) [Other Versions below]

```
/* YourDuino.com Example Software Sketch
 IR Remote Kit Test
 Uses YourDuino.com IR Infrared Remote Control Kit 2
 http://arduino-direct.com/sunshop/index.php?l=product_detail&p=153
 based on code by Ken Shirriff - http://arcfn.com
 Get Library at: https://github.com/shirriff/Arduino-IRremote
 Unzip folder into Libraries. RENAME folder IRremote
 terry@yourduino.com */

/*-----( Import needed libraries )-----*/

#include "IRremote.h"

/*-----( Declare Constants )-----*/
int receiver = 11; // pin 1 of IR receiver to Arduino digital pin 11

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


void setup()   /*----( SETUP: RUNS ONCE )----*/
{
  Serial.begin(9600);
  Serial.println("IR Receiver Raw Data + Button Decode Test");
  irrecv.enableIRIn(); // Start the receiver

}/*--(end setup )---*/


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

  {
//    Serial.println(results.value, HEX);  UN Comment to see raw values
    translateIR();
    irrecv.resume(); // receive the next value
  }
}/* --(end main loop )-- */
```

```
/*-----( Declare User-written Functions )-----*/
void translateIR() // takes action based on IR code received

// describing Car MP3 IR codes

{

  switch(results.value)

  {

  case 0xFFA25D:
    Serial.println(" CH-            ");
    break;

  case 0xFF629D:
    Serial.println(" CH             ");
    break;

  case 0xFFE21D:
    Serial.println(" CH+            ");
    break;

  case 0xFF22DD:
    Serial.println(" PREV           ");
    break;

  case 0xFF02FD:
    Serial.println(" NEXT           ");
    break;

  case 0xFFC23D:
    Serial.println(" PLAY/PAUSE     ");
    break;

  case 0xFFE01F:
    Serial.println(" VOL-           ");
    break;

  case 0xFFA857:
    Serial.println(" VOL+           ");
    break;

  case 0xFF906F:
    Serial.println(" EQ             ");
    break;

  case 0xFF6897:
    Serial.println(" 0              ");
    break;

  case 0xFF9867:
    Serial.println(" 100+           ");
    break;

  case 0xFFB04F:
    Serial.println(" 200+           ");
    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;

  default:
    Serial.println(" other button   ");

  }

  delay(500);


} //END translateIR



/* ( THE END ) */
```

OTHER IR Remote Kit Test Sketches (Click to Download):

IR_Remote_Kit_Numeric.pde
Details Download 3 KB

- Same as example above, but returns a number for the button pressed:

0..9,100,200, Top 9 Buttons encoded 10..18, -1 Bad Code, -2 REPEAT

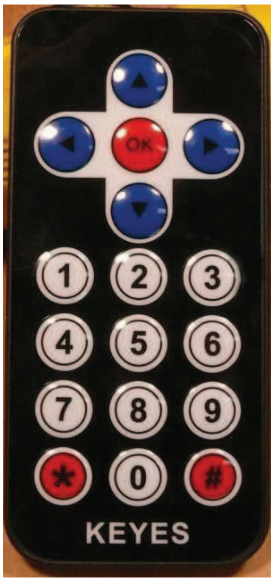
IR_Remote_Kit_Blink.pde
Details Download 3 KB

Blinks Pin 13 number of times according to button number. This may be a place to start when writing your own code to take actions depending on the button.

If you use the IRrecvDemo Sketch (above) and count the 21 buttons from left to right and top to bottom, the codes received are these: (NOTE: Receiving "FFFFFFFF" means "repeat" if you hold the button down.)

1	FFA25D
2	FF629D
3	FFE21D
4	FF22DD
5	FF02FD
6	FFC23D
7	FFE01F
8	FFA857
9	FF906F
10	FF6897
11	FF9867
12	FFB04F
13	FF30CF
14	FF18E7
15	FF7A85
16	FF10EF
17	FF38C7
18	FF5AA5
19	FF42BD
20	FF4AB5
21	FF52AD

Example: MAKER Version Electronic Brick Set IR Remote



The IR Remote supplied with this Set looks like this (Others may also be supplied):

- Based on NEC protocol; Built-in 1 x AG10 battery;
- Remote control range: above 8m;
- Wavelength: 940Nm;
- Frequency: crystal oscillator: 455KHz; IR carrier frequency: 38KHz

This is especially good for remote control of a small robot, using the arrow buttons. Below is an example Software Sketch for this remote. The reported buttons will be **Forward, Left, Right, Reverse** (for the 4 blue button), **OK** for the red 'OK' button, **1 to 0** for the white number buttons, and **'*'** and **'#'** for the bottom red buttons.

(Copy and paste the Sketch below into a blank Arduino IDE Window), Upload to your Arduino and start the Serial Monitor window. Connect the IR receiver to +5V, Ground and the signal to pin 11. If you have the MAKER Version Electronic Brick Starter Set you can just plug in the supplied IR Receiver Brick with one of the 3-pin cables.

```
/* YourDuino.com Example Software Sketch
 Brick Starter Set IR Remote Kit Test
http://yourduino.com/sunshop2/index.php?l=product_detail&p=364
 based on code by Ken Shirriff - http://arcfn.com
 Get Library at: https://github.com/shirriff/Arduino-IRremote
 Unzip folder into Libraries. RENAME folder IRremote
 terry@yourduino.com */

/*-----( Import needed libraries )-----*/

#include "IRremote.h"

/*-----( Declare Constants )-----*/
int receiver = 11; // pin 1 of IR receiver to Arduino digital pin 11

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


void setup()   /*----( SETUP: RUNS ONCE )----*/
{
  Serial.begin(9600);
  Serial.println("YourDuino IR Receiver Button Decode Test");
  Serial.println("Questions: terry@yourduino.com");
  irrecv.enableIRIn(); // Start the receiver

}/*--(end setup )---*/


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

  {
//    Serial.println(results.value, HEX);  UN Comment to see raw values
```

w.wikispaces.com/user/join?goto=https%3A%2F%2Farduino-info.wikispaces.com%2Fspace%2Fjoin) Welcome: Learn more about what Wikispaces has to offer (https://www.wikispaces.com/try/searchbanner). | Ayuda (http://helpcenter.wikispaces.com/) | Iniciar Sesión (https://www.wikispaces.com/site/signin?goto=https%3A%2F%2Farduino-info.wikispaces.com%2FIR-RemoteControl%3FresponseToken%3D0ae6d5

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

/*-----( Declare User-written Functions )-----*/
void translateIR() // takes action based on IR code received

// describing KEYES Remote IR codes

{

  switch(results.value)

  {

  case 0xFF629D: Serial.println(" FORWARD"); break;
  case 0xFF22DD: Serial.println(" LEFT");    break;
  case 0xFF02FD: Serial.println(" -OK-");    break;
  case 0xFFC23D: Serial.println(" RIGHT");   break;
  case 0xFFA857: Serial.println(" REVERSE"); break;
  case 0xFF6897: Serial.println(" 1");    break;
  case 0xFF9867: Serial.println(" 2");    break;
  case 0xFFB04F: Serial.println(" 3");    break;
  case 0xFF30CF: Serial.println(" 4");    break;
  case 0xFF18E7: Serial.println(" 5");    break;
  case 0xFF7A85: Serial.println(" 6");    break;
  case 0xFF10EF: Serial.println(" 7");    break;
  case 0xFF38C7: Serial.println(" 8");    break;
  case 0xFF5AA5: Serial.println(" 9");    break;
  case 0xFF42BD: Serial.println(" *");    break;
  case 0xFF4AB5: Serial.println(" 0");    break;
  case 0xFF52AD: Serial.println(" #");    break;
  case 0xFFFFFFFF: Serial.println(" REPEAT");break;

  default:
    Serial.println(" other button   ");

  }// End Case

  delay(500); // Do not get immediate repeat


} //END translateIR



/* ( THE END ) */
```

(https://www.wikispaces.com/user/view/ray_baker)

Unable to complie MAKER Version Electronic Brick Set IR Remote

ray_baker (https://www.wikispaces.com/user/view/ray_baker) Feb 22, 2016

Arduino: 1.6.7 (Windows 10), Board: "Arduino Nano, ATmega328"

```
C:\Program Files
(x86)\Arduino
\arduino-builder
-dump-prefs
-logger=machine
-hardware
"C:\Program Files
(x86)\Arduino
\hardware" -tools
"C:\Program Files
(x86)\Arduino\tools-
builder" -tools
"C:\Program Files
(x86)\Arduino
\hardware\tools\avr"
-built-in-libraries
"C:\Program Files
(x86)\Arduino
\libraries" -libraries
"C:\WINDOWS
\system32\config
\systemprofile
\Documents\Arduino
\libraries"
-fqbn=arduino:avr:nano:cpu=
-ide-version=10607
-build-path
"C:\Users\ray21
\AppData\Local
\Temp\buildace904fbac00a9
-warnings=none
-prefs=build.warn_data_perc
-verbose
"C:\WINDOWS
\system32\config
\systemprofile
\Documents\Arduino
\keyes_IR_remote
\keyes_IR_remote.ino"
C:\Program Files
(x86)\Arduino
\arduino-builder
-compile
-logger=machine
-hardware
"C:\Program Files
(x86)\Arduino
\hardware" -tools
"C:\Program Files
(x86)\Arduino\tools-
```