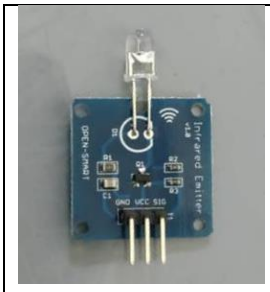
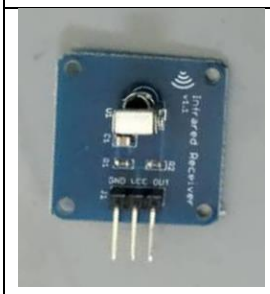
Infrared Transmitter and Infrared Receiver Module.

	The infrared **transmitter module** consists of an infrared diode, and a resistance circuit. It has three pinouts. A ground, a voltage source and a signal pin.
	The infrared **receiver module** consists of an infrared receiver, and a bandpass filter circuit. It has three pinouts. A ground, a voltage source, and a signal pin.

Using with Arduino

To use IR modules with arduino, IRremote library need to be downloaded. The functions are as follows

Receiving

IRrecv irrecv(receivePin)

Create the receiver object, using a name of your choice.

irrecv.enableIRIn()

Begin the receiving process. This will enable the timer interrupt which consumes a small amount of CPU every 50 µs.

irrecv.decode(&results)

Attempt to receive a IR code. Returns true if a code was received, or false if nothing received yet. When a code is received, information is stored into "results".

results.decode_type: Will be one of the following: NEC, SONY, RC5, RC6, or UNKNOWN.

results.value: The actual IR code (0 if type is UNKNOWN)

results.bits: The number of bits used by this code

results.rawbuf: An array of IR pulse times

results.rawlen: The number of items stored in the array

irrecv.resume()

After receiving, this must be called to reset the receiver and prepare it to receive another code.

irrecv.blink13(true)

Enable blinking the LED when during reception. Because you can't see infrared light, blinking the LED can be useful while troubleshooting, or just to give visual feedback.

Transmitting

IRsend irsend;

Create the transmit object. A fixed pin number is always used, depending on which timer the library is utilizing.

irsend.sendNEC(IRcode, numBits);

Send a code in NEC format.

irsend.sendSony(IRcode, numBits);

Send a code in Sony format.

irsend.sendRC5(IRcode, numBits);

Send a code in RC5 format.

irsend.sendRC6(IRcode, numBits);

Send a code in RC6

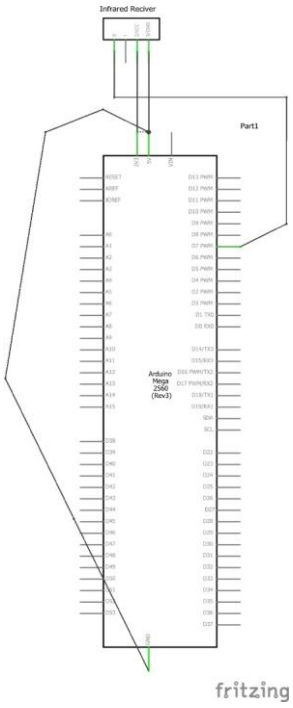
Program for receiver

```
#include <IRremote.h>

const int RECV_PIN = 7;
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);
        irrecv.resume();
  }
}
```



Program for transmitter

```
#include <IRremote.h>

IRsend irsend;
int inPin = 10;
int val = 0;

void setup()
{
 pinMode(inPin, INPUT);
}

void loop() {
     val = digitalRead(inPin);  // read input value
     if (val == LOW)
    {irsend.sendSony(0x68B92, 20); //send 0x0 code (8 bits)
    delay(200);}

}
```