### **Arduino Code**

// Wii Remote IR sensor test sample code by kako http://www.kako.com  
// modified output for Wii-BlobTrack program by RobotFreak http://www.letsmakerobots.com/user/1433  
// modified for http://DFRobot.com by Lumi, Jan. 2014  
  
#include <Wire.h>  
  
int IRsensorAddress = 0xB0;  
//int IRsensorAddress = 0x58;  
int slaveAddress;  
int ledPin = 13;  
boolean ledState = false;  
byte data\_buf[16];  
int i;  
  
int Ix[4];  
int Iy[4];  
int s;  
  
void Write\_2bytes(byte d1, byte d2)  
{  
 Wire.beginTransmission(slaveAddress);  
 Wire.write(d1); Wire.write(d2);  
 Wire.endTransmission();  
}  
  
void setup()  
{  
 slaveAddress = IRsensorAddress >> 1; // This results in 0x21 as the address to pass to TWI  
 Serial.begin(19200);  
 pinMode(ledPin, OUTPUT); // Set the LED pin as output  
 Wire.begin();  
 // IR sensor initialize  
 Write\_2bytes(0x30,0x01); delay(10);  
 Write\_2bytes(0x30,0x08); delay(10);  
 Write\_2bytes(0x06,0x90); delay(10);  
 Write\_2bytes(0x08,0xC0); delay(10);  
 Write\_2bytes(0x1A,0x40); delay(10);  
 Write\_2bytes(0x33,0x33); delay(10);  
 delay(100);  
}  
void loop()  
{  
 ledState = !ledState;  
 if (ledState) { digitalWrite(ledPin,HIGH); } else { digitalWrite(ledPin,LOW); }  
  
 //IR sensor read  
 Wire.beginTransmission(slaveAddress);  
 Wire.write(0x36);  
 Wire.endTransmission();  
  
 Wire.requestFrom(slaveAddress, 16); // Request the 2 byte heading (MSB comes first)  
 for (i=0;i<16;i++) { data\_buf[i]=0; }  
 i=0;  
 while(Wire.available() && i < 16) {   
 data\_buf[i] = Wire.read();  
 i++;  
 }  
  
 Ix[0] = data\_buf[1];  
 Iy[0] = data\_buf[2];  
 s = data\_buf[3];  
 Ix[0] += (s & 0x30) <<4;  
 Iy[0] += (s & 0xC0) <<2;  
  
 Ix[1] = data\_buf[4];  
 Iy[1] = data\_buf[5];  
 s = data\_buf[6];  
 Ix[1] += (s & 0x30) <<4;  
 Iy[1] += (s & 0xC0) <<2;  
  
 Ix[2] = data\_buf[7];  
 Iy[2] = data\_buf[8];  
 s = data\_buf[9];  
 Ix[2] += (s & 0x30) <<4;  
 Iy[2] += (s & 0xC0) <<2;  
  
 Ix[3] = data\_buf[10];  
 Iy[3] = data\_buf[11];  
 s = data\_buf[12];  
 Ix[3] += (s & 0x30) <<4;  
 Iy[3] += (s & 0xC0) <<2;  
  
 for(i=0; i<4; i++)  
 {  
 if (Ix[i] < 1000)  
 Serial.print("");  
 if (Ix[i] < 100)   
 Serial.print("");  
 if (Ix[i] < 10)   
 Serial.print("");  
 Serial.print( int(Ix[i]) );  
 Serial.print(",");  
 if (Iy[i] < 1000)  
 Serial.print("");  
 if (Iy[i] < 100)   
 Serial.print("");  
 if (Iy[i] < 10)   
 Serial.print("");  
 Serial.print( int(Iy[i]) );  
 if (i<3)  
 Serial.print(",");  
 }  
 Serial.println("");  
 delay(15);  
}

### **Processing Code**

// Example by Tom Igoe  
// Modified for http://www.DFRobot.com by Lumi, Jan. 2014  
  
/\*  
 This code should show one colored blob for each detected IR source (max four) at the relative position to the camera.  
\*/  
  
import processing.serial.\*;  
  
int lf = 10; // Linefeed in ASCII  
String myString = null;  
Serial myPort; // The serial port  
  
void setup() {  
 // List all the available serial ports  
 println(Serial.list());  
 // Open the port you are using at the rate you want:  
 myPort = new Serial(this, Serial.list()[0], 19200);  
 myPort.clear();  
 // Throw out the first reading, in case we started reading   
 // in the middle of a string from the sender.  
 myString = myPort.readStringUntil(lf);  
 myString = null;  
 size(800,800);  
 //frameRate(30);  
}  
  
void draw() {  
 background(77);  
 //while (myPort.available() > 0) {  
 myString = myPort.readStringUntil(lf);  
 if (myString != null) {  
 int[] output = int (split(myString, ','));  
  
 println(myString); // display the incoming string  
   
 int xx = output[0];  
 int yy = output[1];  
   
 int ww = output[2];  
 int zz = output[3];  
   
 int xxx = output[4];  
 int yyy = output[5];  
   
 int www = output[6];  
 int zzz = output[7];  
  
 ellipseMode(RADIUS); // Set ellipseMode to RADIUS  
 fill(255, 0, 0); // Set fill to white  
 ellipse(xx, yy, 20, 20);  
 ellipseMode(RADIUS); // Set ellipseMode to RADIUS  
 fill(0, 255, 0); // Set fill to white  
 ellipse(ww, zz, 20, 20);  
   
 ellipseMode(RADIUS); // Set ellipseMode to RADIUS  
 fill(0, 0, 255); // Set fill to white  
 ellipse(xxx, yyy, 20, 20);  
 ellipseMode(RADIUS); // Set ellipseMode to RADIUS  
 fill(255); // Set fill to white  
 ellipse(www, zzz, 20, 20);  
  
 }  
}