Application Design： Catch the Ball

Key Points

1. Learn IIC for communication between two MCUs
2. Make your robot pursue an infrared ball
3. Equipment needed：miniQ v2.0， MicroUSB Cable，Upperdeck for MiniQ, Infrared transmitters and receivers, Proto board, soldering station, soldering iron.

Tutorial

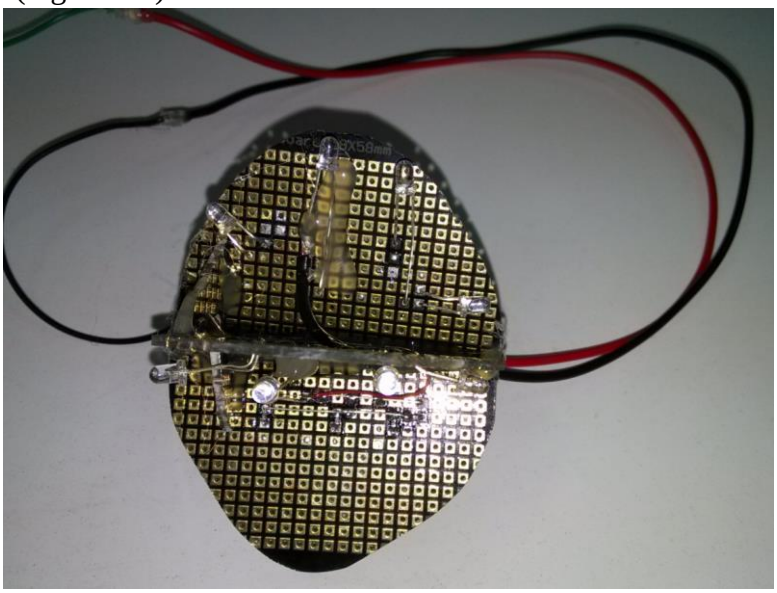
We have learned much about the program and circuit about your robot. And we feel that stopping this series of tutorial by obstacle avoidance or line tracking is not enough. So let’s try to make a fun project.

And we don’t need our robot to pursue a real rolling ball, so we use an infrared ball, thus the robot can only follow your ball. Thus the robot is absolutely yours. And things may be a little difficult as we need to make an infrared ball first. The ball is not easy to make but the circuit is easy---just a lot infrared LED connect in parallel.

This is a kind of infrared ball:

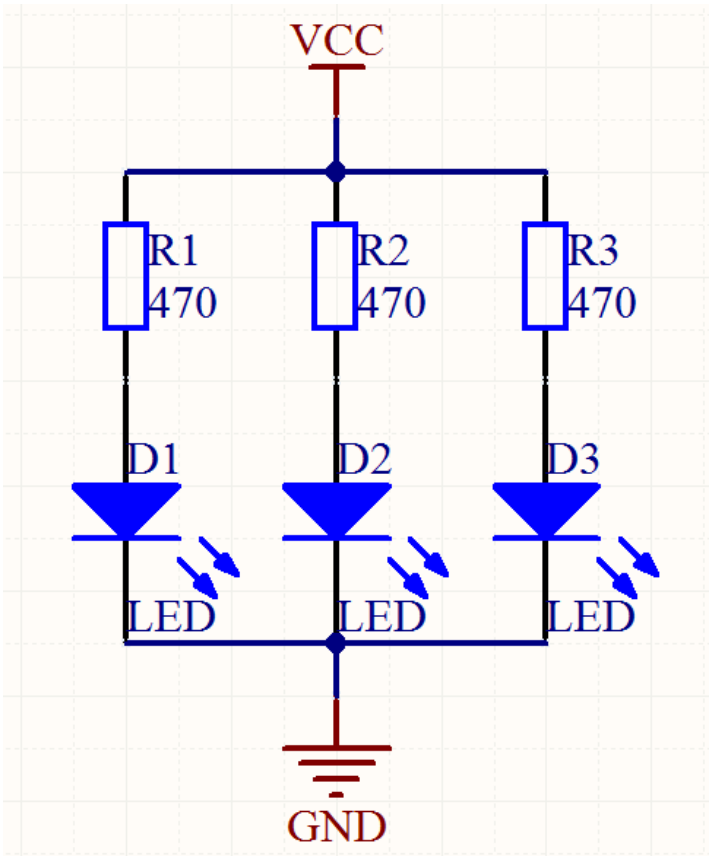
1 Infrared Ball

And a little(big in fact) alteration of the ball I made:



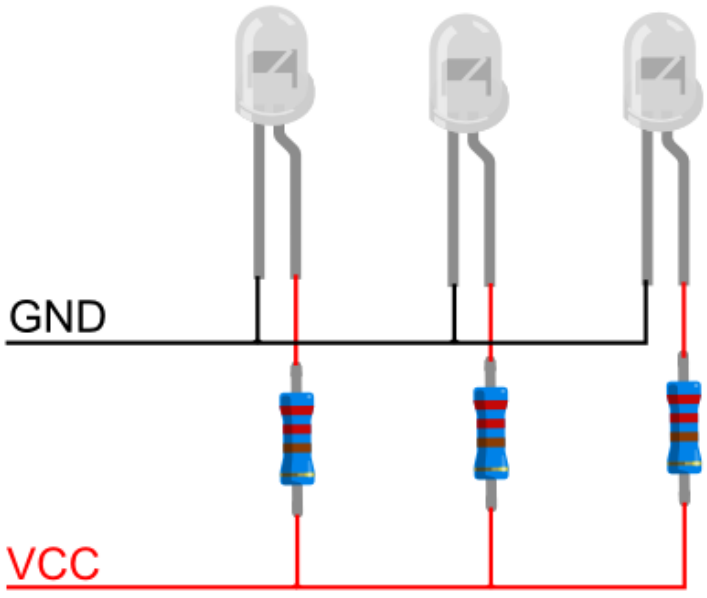
2 Ball I made

That doesn’t seem like a ball, but it can work. And you can alse do some decoration. For the circuit, as simple as usual:



3. Circuit of infrared leds

R1，R2，R3 are resisitors of 470Ω，D1，D2，D3 are infrared led. And we aslo have another picture for check your circuit:



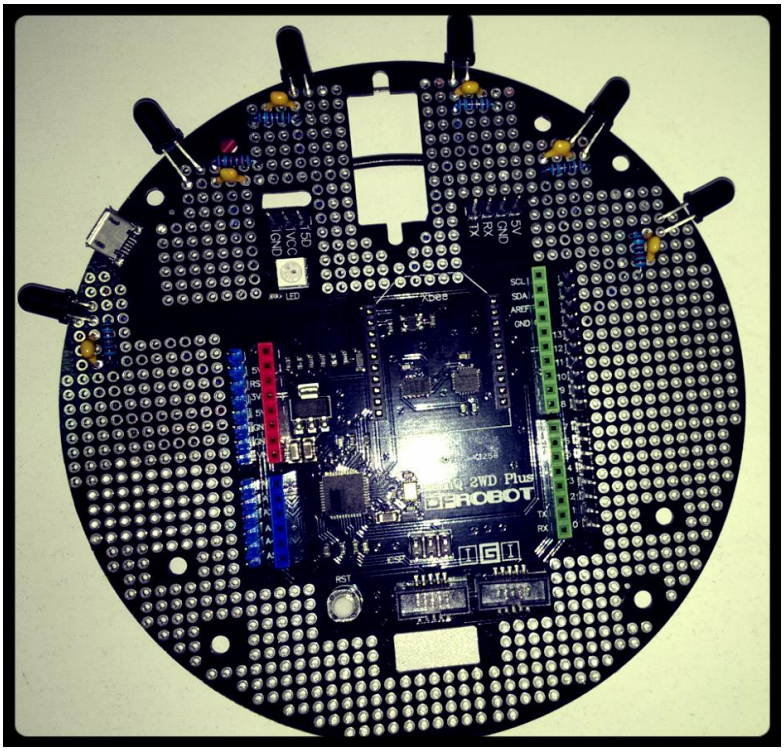
4 Infrared LEDs connection diagram

Power supply can be 4-6V, you can choose one to fit your situation, the resistor

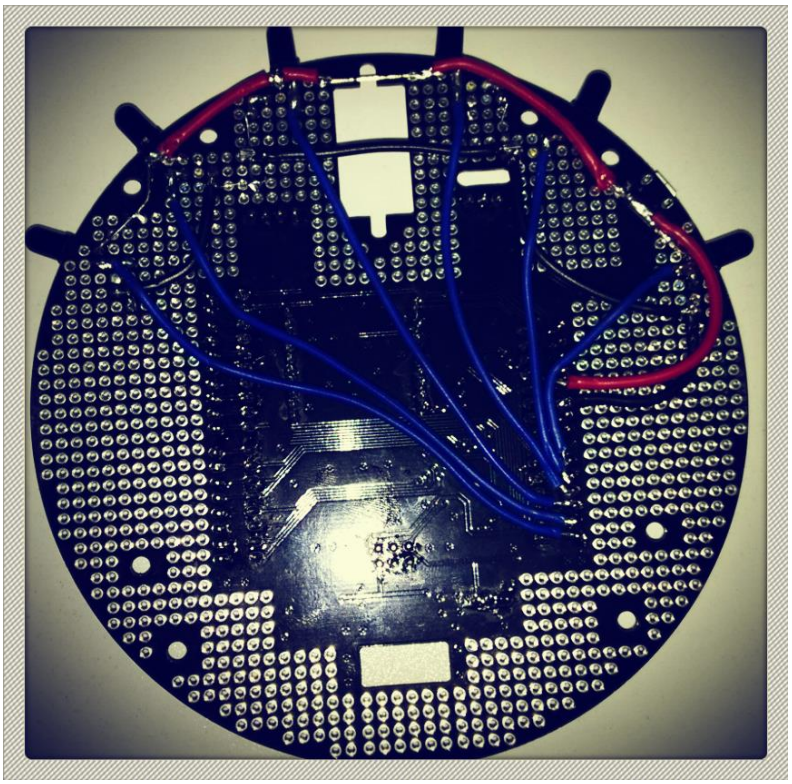
can be 470 to 1kΩ, and 470 is recommended.

This board is designed for miniQ, we can see it as a round board consist of Leonardo and external proto board. It makes Leonardo more convenient to use and more customized for miniQ.

See what we will do for the board:



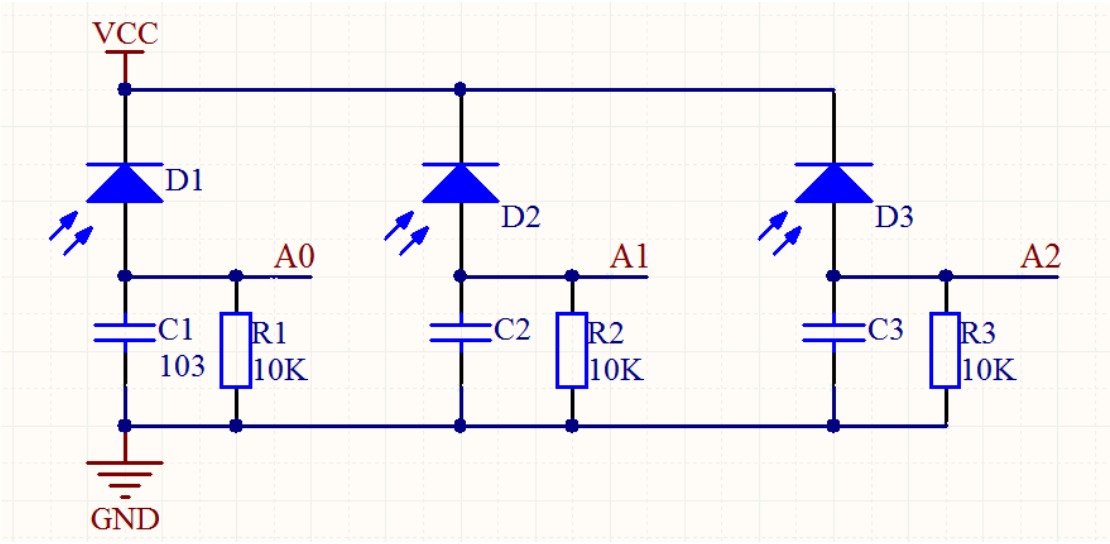
5 Top board with infrared receivers



6 Top board with infrared receivers

In this step we will get enough soldering experience, start by soldering the receivers. That is really a good soldering experience for beginners. And be careful about the hot electric soldering iron. When finish soldering, check your circuit twice for bridged solder spots, like ground connected to power, to ensure your components won't be destroyed.

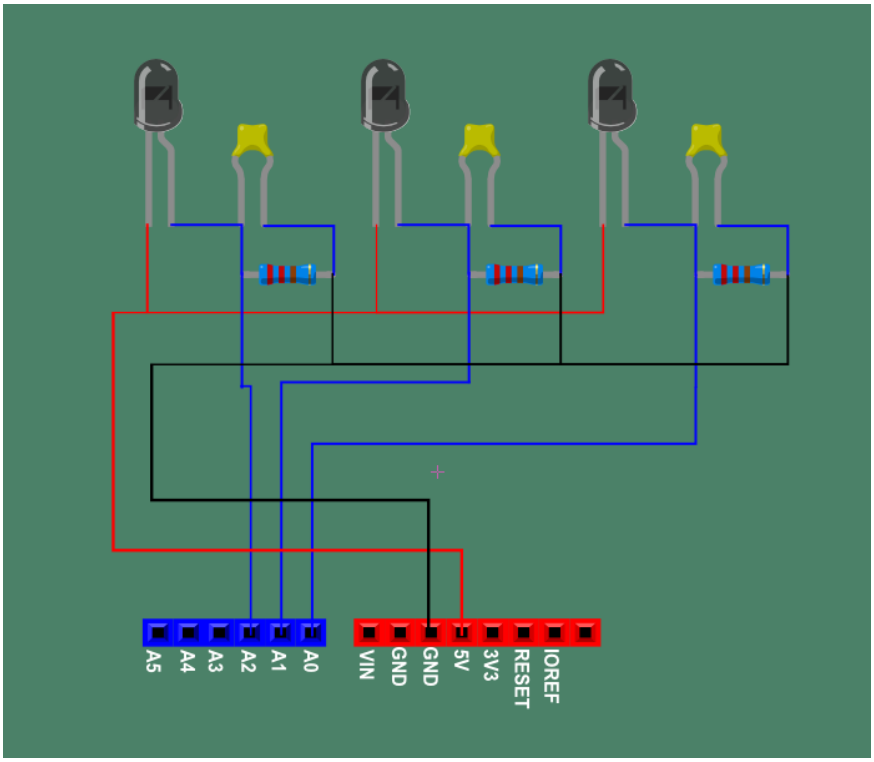
The schematic diagram:



7 schematic diagram of infrared receivers

D1，D2，D3 are infrared receivers，A0，A1，A2 connect analog ports in arduino A1,A2 and A3. All the 6 receivers use the some way so the left ones you connect yourselves.

If you don't like schematic diagram, you can check your soldering with this diagram:

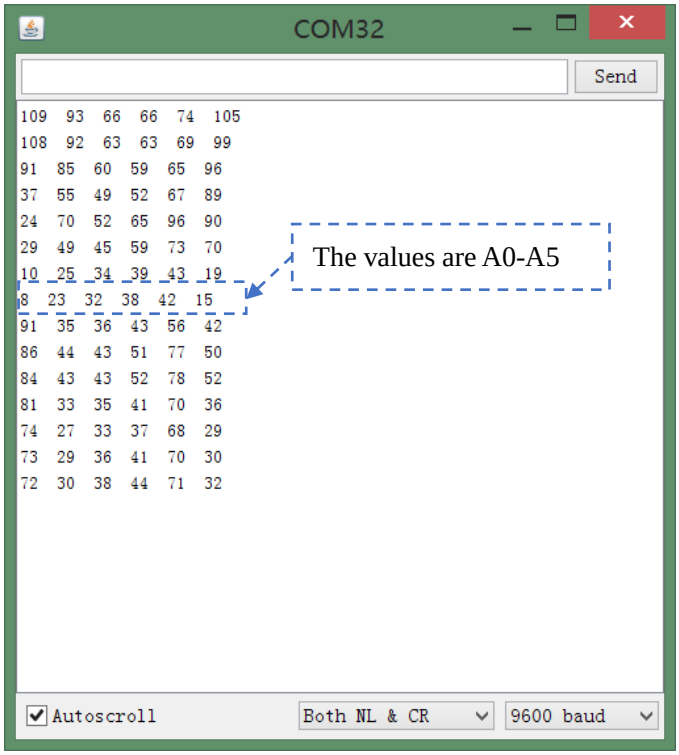


8. Sketch map of infrared receivers

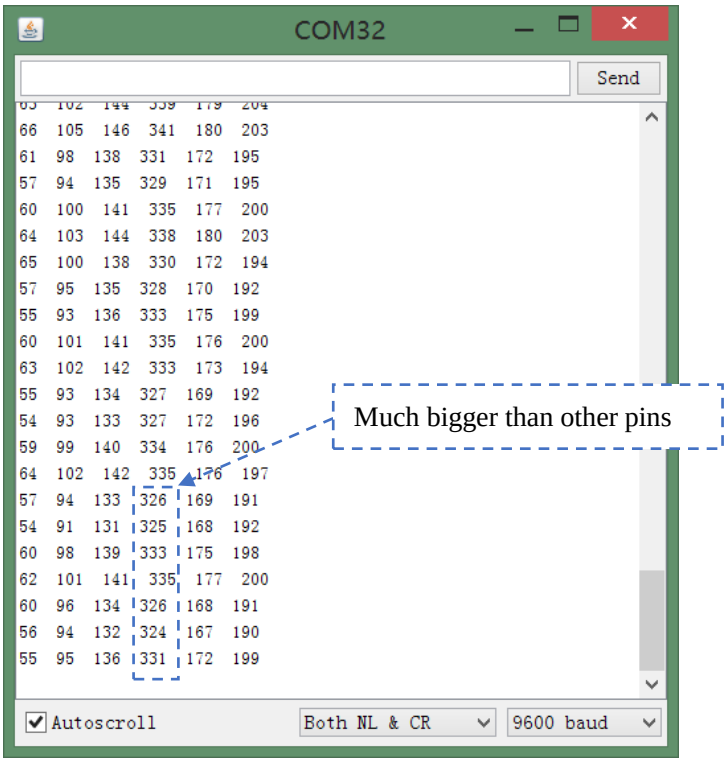
OK, after soldering the receivers, now check it：
Download the code:

```
void setup()
 {
    Serial.begin(9600);//baud rate: 9600
 }

 void loop()
 {
    for(int i=0;i<6;i++)
    {
       int x=analogRead(i);    //read the analog value of pin “i”, “i” is 0~5
       Serial.print(x);              //print the value read
       Serial.print("    ");
    }
    Serial.println();
    delay(500);                        //delay 500ms
 }
```

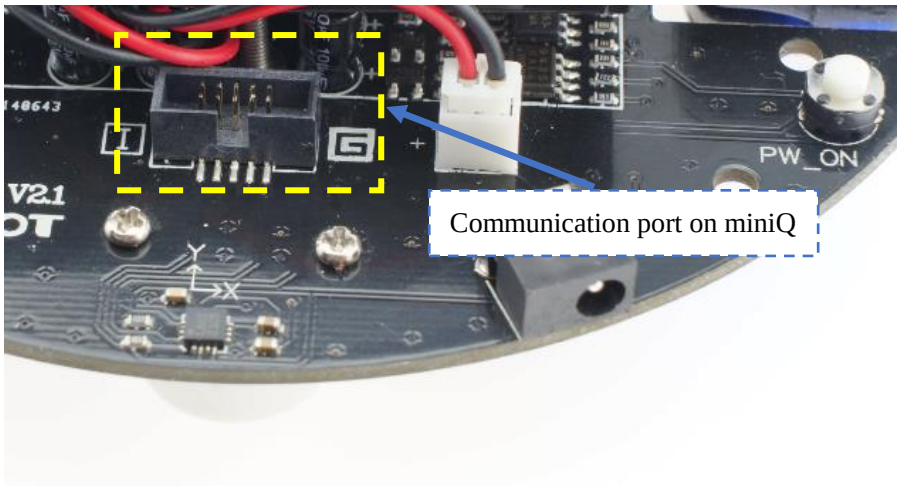


9. Value get in normal situation



10. Put your infrared ball in front of your board

If your board performanced well, you can install the board on your miniQ. Just connect them use this port:



11.The connection port

We need to download two programs into miniQ and the board. “Bottom.ino” for miniQ and “Top.ino” for the board.

Code Analysis

- Top board:

✧ Function for send command

```
void Send_Command(char command,int num)
 {
  Wire.beginTransmission(Address_Bottom); //send data to address 2
  Wire.write(command);              //send---action command
  Wire.write(num);              //send data----value
  Wire.endTransmission();       // stop transmission
  Serial.println(num);//for debug, can be deleted
 }
```

✧ Get the direction of the ball

```
for(int i=0;i<6;i++)              //Find the biggest value and which sensor gives the value
  {
       if(Distance[i]>Dist_Max)
          {
             Dist_Max=Distance[i];
             Num_Max=i;
          }
    Serial.print(Distance[i]);
    Serial.print(" ");
  }
```

✧ When the biggest value is more than 100, send the command

```
if(Dist_Max>100)
  {
        if(Num_Max<2) {Send_Command(cmd[2],100);}//if the pin is 0 or 1, let miniQ turn Left
   else if(Num_Max==4 || Num_Max==5)
 {
     Send_Command(cmd[3],100);
 }//if the pin is 4 or 5, send “R100”
   else if(Num_Max==2 || Num_Max==3)
          {
             if(Dist_Max>800)//if the value is too big(get the ball)
             {
               Send_Command(cmd[4],100);
               Serial.println(cmd[4]);delay(200);//S100 for Stop
             }
             else
             {
                Send_Command(cmd[0],100);Serial.println(cmd[0]);   //if distance is too far, go forward
```

```
                }
              }
    }
    else
    {
        Send_Command(cmd[4],80);Serial.println(cmd[2]);
    }
```

- Code for miniQ：

✧ Main function：

```
void loop()
{
    Motor();//Control motors
    Color();//Control RGB LED
}
```

✧ Function for control LED

```
void Color()//Read the command, then action
{
    switch (Cmd_Str)
    {
        case 'F':    strip.setPixelColor(0, 0xbb0000);        strip.show(); break;
        case 'B':    strip.setPixelColor(0, 0x0000bb);        strip.show(); break;
        case 'L':    strip.setPixelColor(0, 0x00bb00);        strip.show(); break;
        case 'R':    strip.setPixelColor(0, 0x550055);        strip.show(); break;
        case 'S':    strip.setPixelColor(0, 0x005555);        strip.show(); break;
        default: break;
    }
}
```

✧ Function for motors

```
void Motor()//Read the command, then action
{
    switch (Cmd_Str)
    {
        case 'F': Forward(Cmd_Num,Cmd_Num);          break;
        case 'B': Back(Cmd_Num,Cmd_Num);             break;
        case 'L': Turn_Left(Cmd_Num,Cmd_Num);        break;
```

```
        case 'R': Turn_Right(Cmd_Num,Cmd_Num);       break;
        case 'S': Stop();                                    break;
        default: break;
    }
}
```

✧ **Action for command receive**

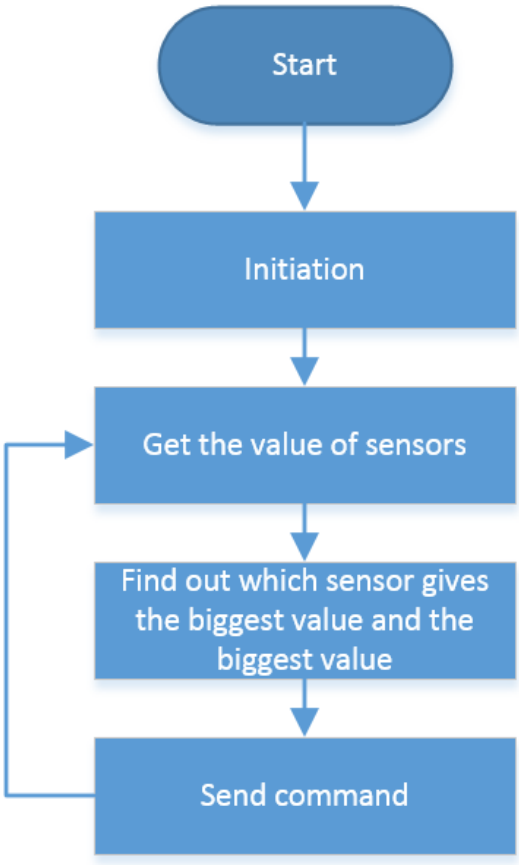
```
void receiveEvent(int howMany)//Receive interrupt
{
  while(Wire.available()>1) // loop through all but the last
  {
    Cmd_Str = Wire.read(); // receive byte as a character
    Serial.print(Cmd_Str);         // print the character
  }
  Serial.print("    ");
  Cmd_Num = Wire.read();       // receive byte as an integer
  Serial.println(Cmd_Num);              // print the integer
}
```

How Does It Work?

In this project, we use two Arduino boards, and each of them do their own job. One for find the ball and another for control the motors. You may be wondering that why not only use one MiniQ without the “Leonardo” board? One reason is the port of MiniQ don’t connect some pin header connecters for us and one more import point is the pins is nearly used up. So it is better to use another board to collect the information.

The “Leonardo” will send command to miniQ through IIC, and miniQ will have different reaction when receive different command. Cause each part do their own job, so when an error happen, it is easy to find the reason and then solve it.

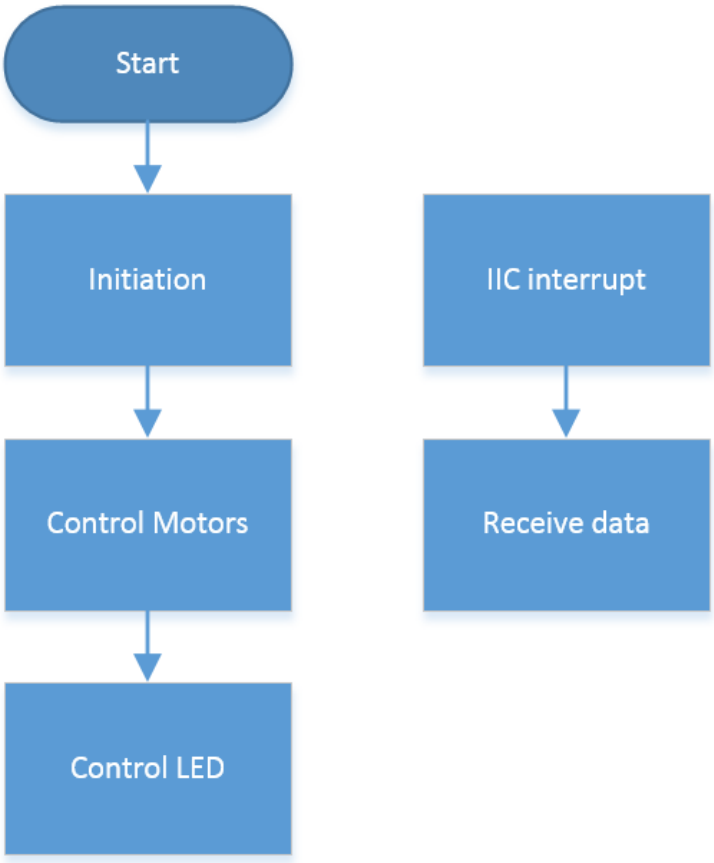
Let’s see the algorithm in “Leonardo”:



12. Flow chart for top board

From the flow chart, we can see “Leonardo” need to get information of every sensor and find the which sensor gives the biggest value, then send a command depend on the value. It didn’t need to consider how to driver the motors ans LEDs.

And the flow chart of miniQ:



13. The flow chart of miniQ

All the action of miniQ will depend on the command of “Leonardo”. So MiniQ do not need to know the feedback of sensors.

Here we will learn how to communicate each other, the answer is called IIC protocol. We use IIC to send data to each other, but how the other one knows which action should it finish? Take a simple example, the expansion shield want miniQ to turn left at speed of 100, it just need to send “L” and “100” to miniQ, and for miniQ, many data will be received, if the data is “H999”, that’s not effective, just ignore it.

In this way, you can work with your friends and it is easy to distinguish which one programs wrong. That is really a good way to finish big project, right?