```
1 //============================================================================
2 // Name        : BB-pixy.cpp
3 // Author      : Jaspreet Singh
4 // Version     :
5 // Copyright   : Your copyright notice
6 // Description : Hello World in C++, Ansi-style
7 //============================================================================
8
9 #include <iostream>
10 #include"Pixy2BBB.h"
11 #include"uart.h"
12 #include"TPixy2.h"
13 using namespace std;
14
15 Uart lpc_link;
16 Uart Bluetooth;
17 Pixy2BBB pixy;
18
19
20 int x;
21 int y;
22 int sig;
23 int x_min=70;
24 int x_max=200;
25 unsigned int maxArea=8000;
26 unsigned int minArea=1000;
27 unsigned int width;
28 unsigned int height;
29 unsigned int area;
30 unsigned int newarea;
31 int i=0;
32 //uint16_t blocks;
33 int mySig, Sig_Status=0, Mode_Status=0;
34
35 /* Characters used for data storage */
36
37 unsigned char bt_data,Mode='n',Color='n',Manual_inst,j,Track_Mode=1,Find=1;
38
39
40
41 int Track(char tsig)
42 {
43
44       pixy.ccc.getBlocks();  //receive data from pixy
45
46           if(pixy.ccc.numBlocks)
47          {
48               usleep(10000);
49               if(Find==1)
50               {
51                   lpc_link.send("S");
52                   Find=0;
53               }
54              sig = pixy.ccc.blocks[i].m_signature;    //get object's signature
55              x = pixy.ccc.blocks[i].m_x;                    //get x position
56              y = pixy.ccc.blocks[i].m_y;                    //get y position
57              width = pixy.ccc.blocks[i].m_width;            //get width
58              height = pixy.ccc.blocks[i].m_height;          //get height
59            //  printf("sig = %d    x= %d y= %d  width = %d height= %d \n
  area=%d",sig,x,y,width,height,width*height);
60
61
62
63              if(sig==tsig)
64              {
65                    newarea= width * height;
66
67                    printf("newarea %d\n",newarea);
68
69                    if(x<x_min )
70                    {
71                        lpc_link.send("L");                        //Send command to lpc to turn left
```

```
72                      //printf("Left\n");
73                      usleep(100000);
74
75                  }
76                  if(x>x_max )
77                  {
78                      lpc_link.send("R");                         //Send command to lpc to turn right
79                      //printf("Right\n");
80                      usleep(100000);
81                  }
82                  if(newarea<7000)
83                  {
84                      lpc_link.send("M");                         //Send command to lpc to Move
85                      //printf("move\n");
86                      usleep(100000);
87                  }
88                  else if (newarea>maxArea)
89                  {
90                      lpc_link.send("B");                         //Send command to lpc to Move in reverse
  direction
91                      //printf("back\n");
92                      usleep(100000);
93                  }
94               else
95              {
96                  lpc_link.send("S");                         //Send command to lpc to Stop
97                  printf("stop\n");
98                  usleep(300000);
99              }
100         }}
101     else
102     {
103         usleep(300000);
104         lpc_link.send("F");                                 //Send command to lpc to Find object
105         usleep(300000);
106         Find=1;
107     }
108
109     return 0;
110 }
111
112
113 int main()
114 {
115
116     pixy.init();                                            //Initializing Pixy2
117
118
119
120     lpc_link.Init(UART04,115200);                           //Initializing Uart
121     Bluetooth.Init(UART01,9600);                            //Initializing Uart
122
123
124 begining:
125     pixy.setLamp(1,0);
126     Mode_Status=0;
127     Sig_Status=0;
128     lpc_link.send("S");
129     Bluetooth.send("Welcome\n");
130     usleep(900000);
131     Bluetooth.send("Please Select the mode \n");
132     usleep(900000);
133     Bluetooth.send("A for automatic and M for manual \n");
134     usleep(100000);
135     pixy.setLamp(0,0);
136
137     while(Mode_Status==0)
138     {
139         while(Bluetooth.recieve(&Mode)<2);
140
141         //p=Bluetooth.recieve(&Mode);
142         //printf("d%\n",p);
```

```
143
144         if(Mode=='A')
145         {
146             Bluetooth.send("Please Select the color to track \n");
147             usleep(900000);
148             Bluetooth.send("Options- P for Purple Ball \n G for Green Ball \n");
149             usleep(900000);
150
151             while(Sig_Status==0)
152             {
153
154                 usleep(1000);
155                 while(Bluetooth.recieve(&Color)<2);
156                 printf("%d\n",p);
157                 pixy.setLamp(0,0);
158                 if(Color=='P')
159                 {
160                     mySig=1;
161                     Sig_Status=1;
162                     Mode_Status=1;
163                 }
164
165                 else if(Color=='Q')
166                 {
167                     goto begining;
168                 }
169                 else
170                 {
171                     Bluetooth.send("Enter a valid Option\n");
172                     //Sig_Status=0;
173                 }
174             }
175
176         }
177
178         if(Mode=='M')
179         {
180             Bluetooth.send("Manual Mode\n");
181             while(1)
182             {
183                 Bluetooth.recieve(&Manual_inst);
184                 if(Manual_inst=='M')
185                 {
186                     lpc_link.send("M");
187                     usleep(400000);
188                     //printf("ffff");
189                 }
190                 else if(Manual_inst=='B')
191                     {
192                     lpc_link.send("B");
193                     usleep(400000);
194                           }
195                 else if(Manual_inst=='L')
196                 {
197                     lpc_link.send("L");
198                     usleep(400000);
199                 }
200                 else if(Manual_inst=='R')
201                 {
202                     lpc_link.send("R");
203                     usleep(400000);
204
205                 }
206                 else if(Manual_inst=='S')
207                 {
208                     lpc_link.send("S");
209                     usleep(400000);
210                 }
211                 else if(Manual_inst=='F')
212                 {
213                     lpc_link.send("F");
214                     usleep(400000);
```

```
                                }
            else if(Manual_inst=='Q')
            {
                goto begining;
            }
            usleep(200);
        }
    }
    else if(Mode=='Q')
        {
            goto begining;
        }

    else
        {
            Bluetooth.send("Please Choose an valid option \n");
        }


}


Bluetooth.send("Tracking \n");
//pixy.setLamp(1,0);
while(1)
{

    Bluetooth.recieve(&j);

    Track(mySig);
    if(j=='Q')
    {
        lpc_link.send("S");
        usleep(30000);
        goto begining;
    }

}


return 0;
}

```