

1	#include <Servo.h>
2	//defining Servos
3	Servo servohori;
4	int servoh = 0;
5	int servohLimitHigh = 160;
6	int servohLimitLow = 20;
7	
8	Servo servoverti;
9	int servov = 0;
10	int servovLimitHigh = 160;
11	int servovLimitLow = 20;
12	//Assigning LDRs
13	int ldrtopl = 2; //top left LDR green
14	int ldrtopr = 1; //top right LDR yellow
15	int ldrbotl = 3; // bottom left LDR blue
16	int ldrbotr = 0; // bottom right LDR orange
17	
18	void setup ()
19	{
20	servohori.attach(10);
21	servohori.write(0);
22	servoverti.attach(9);
23	servoverti.write(0);
24	delay(500);
25	}
26	
27	void loop()
28	{
29	servoh = servohori.read();
30	servov = servoverti.read();
31	//capturing analog values of each LDR
32	int topl = analogRead(ldrtopl);
33	int topr = analogRead(ldrtopr);
34	int botl = analogRead(ldrbotl);
35	int botr = analogRead(ldrbotr);
36	// calculating average
37	int avgtop = (topl + topr) / 2; //average of top LDRs
38	int avgbot = (botl + botr) / 2; //average of bottom LDRs
39	int avgleft = (topl + botl) / 2; //average of left LDRs
40	int avgright = (topr + botr) / 2; //average of right LDRs
41	
42	if (avgtop < avgbot)
43	{
44	servoverti.write(servov +1);
45	if (servov > servovLimitHigh)
46	{
47	servov = servovLimitHigh;
48	}
49	delay(10);
50	}

51	else if (avgbot < avgtop)
52	{
53	servoverti.write(servov -1);
54	if (servov < servovLimitLow)
55	{
56	servov = servovLimitLow;
57	}
58	delay(10);
59	}
60	else
61	{
62	servoverti.write(servov);
63	}
64	
65	if (avgleft > avgright)
66	{
67	servohori.write(servoh +1);
68	if (servoh > servohLimitHigh)
69	{
70	servoh = servohLimitHigh;
71	}
72	delay(10);
73	}
74	else if (avgright > avgleft)
75	{
76	servohori.write(servoh -1);
77	if (servoh < servohLimitLow)
78	{
79	servoh = servohLimitLow;
80	}
81	delay(10);
82	}
83	else
84	{
85	servohori.write(servoh);
86	}
87	delay(50);
88	}