1. #include <reg52.h>
3. sbit IN1 = P1^0;
4. sbit IN2 = P1^1;
5. sbit ENA = P1^2;
6. sbit IN3 = P1^3;
7. sbit IN4 = P1^4;
8. sbit ENB = P1^5;
10. sbit left1 = P0^0;
11. sbit left2 = P0^1;
12. sbit right1 = P0^2;
13. sbit right2 = P0^3;
15. unsigned char zkb1 = 0;
16. unsigned char zkb2 = 0;
17. unsigned char t = 0;

20. void init()
21. {
22. EA = 1;
23. TMOD |= 0x01;
24. TH0 =(65536 - 100) / 256;
25. TL0 = (65536 - 100)% 256;
26. ET0 = 1;
27. TR0 = 1;
28. }
30. void timer0() interrupt 1
31. {
33. TH0 = (65536 - 100) / 256;
34. TL0 = (65536 - 100) % 256;
36. if(t < zkb1)
37. {
38. //      if(t % 2 == 0)
39. //      {
40. //          ENA = 1;
41. //      }
42. //      else
43. //      {
44. //          ENA = 0;
45. //      }
46. ENA = 1;
47. }
48. else
49. {
50. ENA = 0;
51. }
52. if(t < zkb2)
53. {
54. //      if(t%2 == 0)
55. //      {
56. //          ENB = 1;
57. //      }
58. //      else
59. //      {
60. //          ENB = 0;
61. //      }
62. ENB =1;
63. }
64. else
65. {
66. ENB = 0;
67. }
68. ++t;
69. if(t >= 50)
70. {
71. t = 0;
72. }


76. }
78. void turn\_left1()
79. {
80. zkb1 = 13;
81. zkb2 = 0;
83. }
85. void turn\_left2()
86. {
88. zkb1 = 13;
89. zkb2 = 0;
90. }
92. void turn\_right1()
93. {
94. zkb1 = 0;
95. zkb2 = 13;
96. }
98. void turn\_right2()
99. {
100. zkb1 = 0;
101. zkb2 = 13;
102. }
104. void qianjin()
105. {
106. zkb1 = 8;
107. zkb2 = 8;
108. }
109. void xunji()
110. {
111. unsigned char flag;
112. if((right2 == 0)&&(right1 == 0)&&(left1 == 0)&&(left2 == 0))
113. {
114. flag = 0;
115. }
116. if((right2 == 0)&&(right1 ==1)&&(left1 == 0)&&(left2 == 0))
117. {
118. flag = 1;
119. }
120. if((right2 == 0)&&(right1 == 0)&&(left1 == 1)&&(left2 == 0))
121. {
122. flag = 2;
123. }
124. if((right2 == 0)&&(right1 == 0)&&(left1 == 0)&&(left2 == 1))
125. {
126. flag = 3;
127. }
128. if((right2 == 1)&&(right1 == 0)&&(left1 == 0)&&(left2 == 0))
129. {
130. flag = 4;
131. }
132. switch(flag)
133. {
134. case 0:qianjin();break;
136. case 1:turn\_right1();break;
138. case 2:turn\_left1();break;
140. case 3:turn\_left2();break;
142. case 4:turn\_right2();break;
144. default:qianjin();break;
145. }
146. }
148. void main()
149. {
150. init();
151. //  zkb1 = 40;
152. //  zkb2 = 30;
153. IN1 = 0;
154. IN2 = 1;
155. IN3 = 0;
156. IN4 = 1;
157. while(1)
158. {
160. xunji();
162. }
163. }

P1是用来控制电机的端口，IN~控制前进后退，EN~控制小车能不能走，P0是红外接收的口，为1时表示探测到黑线